

# EXHIBIT 15

# EXHIBIT H

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

**Chart Detailing Defendant's Infringement of U.S. Patent No. 9,971,678**

*Wapp Tech Ltd. & Wapp Tech Corp. v. J.P. Morgan Chase Bank, N.A.*

Case No. 4:23-cv-01137 (E.D. Tex.)

The Accused Instrumentalities include Xcode Developer Tools from Apple Inc. that Defendant uses to test applications for iOS mobile devices. Xcode Developer Tools include tools such as the Xcode integrated development environment ("IDE"), Network Link Conditioner, Instruments, and Simulator.

Based on the information presently available to them, Plaintiffs Wapp Tech Limited Partnership and Wapp Tech Corp. ("Wapp" or "Plaintiffs") are informed and believe that Defendant directly and indirectly infringes U.S. Patent No. 9,971,678 (the "'678 Patent"). Defendant directly infringes the '678 Patent when its employees, agents, and/or representatives use the Accused Instrumentalities to test applications for mobile devices. Upon information and belief, to the extent Defendant uses third parties in the testing process, it indirectly infringes the '678 Patent by actively inducing the direct infringement of third parties contracted to use the Accused Instrumentalities to test applications for mobile devices on Defendant's behalf.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

**TABLE OF CONTENTS**

CLAIM 1 .....	5
[1pre] A system for testing an application for a mobile device comprising: .....	5
1. Network Link Conditioner on iOS Device (e.g., iPhone 12) .....	10
2. Network Link Conditioner on MacOS Computer/Xcode .....	13
<u>[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;.....</u>	<u>19</u>
1. Xcode Developer Tools Emulation/Simulation Overview .....	19
2. Xcode Developer Tool Display Overview .....	39
[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.....	61
CLAIM 2 .....	70
[2] The system of claim 1, wherein the software is configured to enable a user to select from one or more connection simulations for testing how well mobile content performs on the mobile device. ....	70
1. Network Link Conditioner on iOS.....	71
2. Network Link Conditioner on MacOS.....	73
CLAIM 3 .....	74
[3] The system of claim 2, wherein the one or more connection simulations are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation. ....	74

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

CLAIM 4 .....	78
[4] The system of claim 2, wherein the connection simulation includes one or more profiles. ....	78
1. Network Link Conditioner on iOS.....	79
2. Network Link Conditioner on MacOS.....	81
CLAIM 5 .....	82
[5] The system of claim 4, wherein the profiles include preset profiles. ....	82
CLAIM 6 .....	83
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles. ....	83
CLAIM 7 .....	97
[7] The system of claim 4, wherein the profiles are configured to enable a user to create custom profiles. ....	97
CLAIM 9 .....	98
[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments. ....	98
CLAIM 21 .....	107
[21] The system of claim 1, wherein the software is further configured to display data to identify application performance. ....	107
CLAIM 22 .....	111

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[22] The system of claim 21, wherein the user can manage the data. .... 111

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

**Claim 1**

**[1pre] A system for testing an application for a mobile device comprising:**

The Accused Instrumentalities, such as Apple's Xcode Developer Tools (including at least Xcode IDE, Network Link Conditioner, Instruments, and Simulator), are a system for testing an application for a mobile device. Defendant tests its mobile banking applications through its use of the Accused Instrumentalities by, for example, executing compiled source code for the application and monitoring the execution of the compiled source code and creating and executing tests (including but not limited to through the use of unit tests).

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

## Xcode

Build, **test**, and submit your app with Apple's integrated development environment.

### Overview

Xcode is a suite of tools developers use to build apps for Apple platforms. Use Xcode to manage your entire development workflow — from creating your app to **testing**, optimizing, and submitting it to the App Store.

<https://developer.apple.com/documentation/Xcode/>, accessed on 2024/03/13 (annotated)



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

Xcode also includes several development tools to help you rapidly prototype and test your app. Use Simulator for rapid prototyping and testing your app in a simulated environment when a real device isn't available. Use Instruments to profile and analyze your app, improve performance, and investigate system resource usage. Construct 3D content with Reality Composer, train custom machine learning models with Create ML, and identify areas of your app that aren't accessible with Accessibility Inspector.

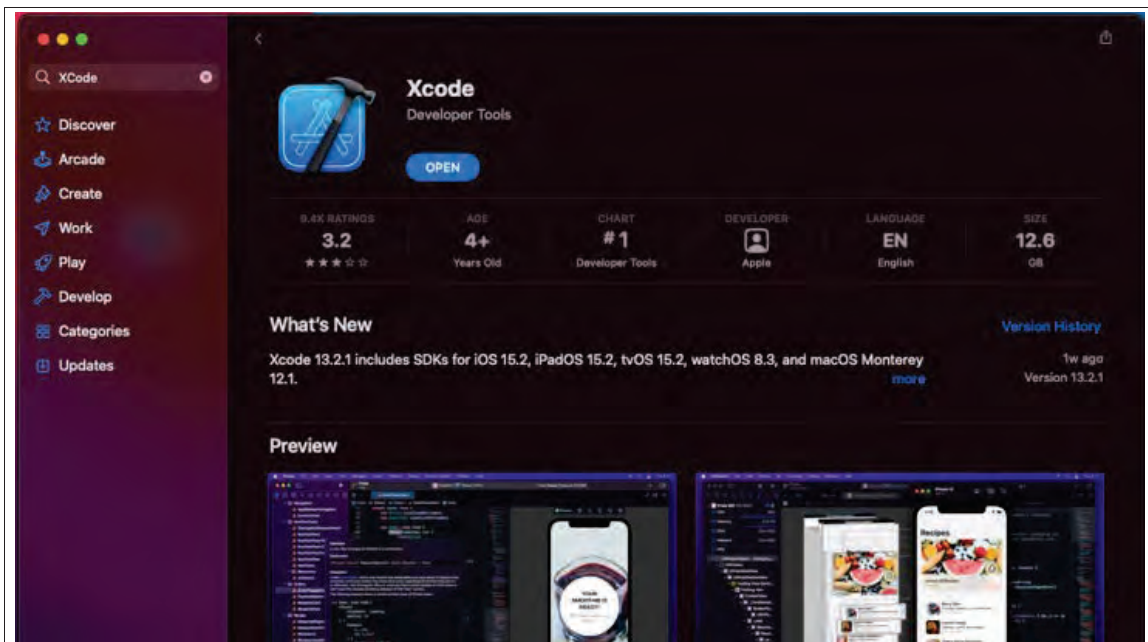
<https://developer.apple.com/documentation/Xcode/>, accessed on 2024/03/13 (annotated).

Xcode IDE is Apple's integrated development environment ("IDE")—it enables developers to test iOS applications by executing compiled source code for the application and monitoring the execution of the compiled source code. Xcode IDE itself runs on a computer running MacOS operating system. Applications tested through Xcode are either tested on a physical iOS device (such as an iPhone), or tested on the computer running Xcode using a simulator. Simulators for various iOS devices are included with Xcode, along with Instruments used to measure the performance of the application being tested.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

On information and belief, Defendant uses Apple's Xcode Developer Tools installed from a source such as Apple's App Store, shown below:

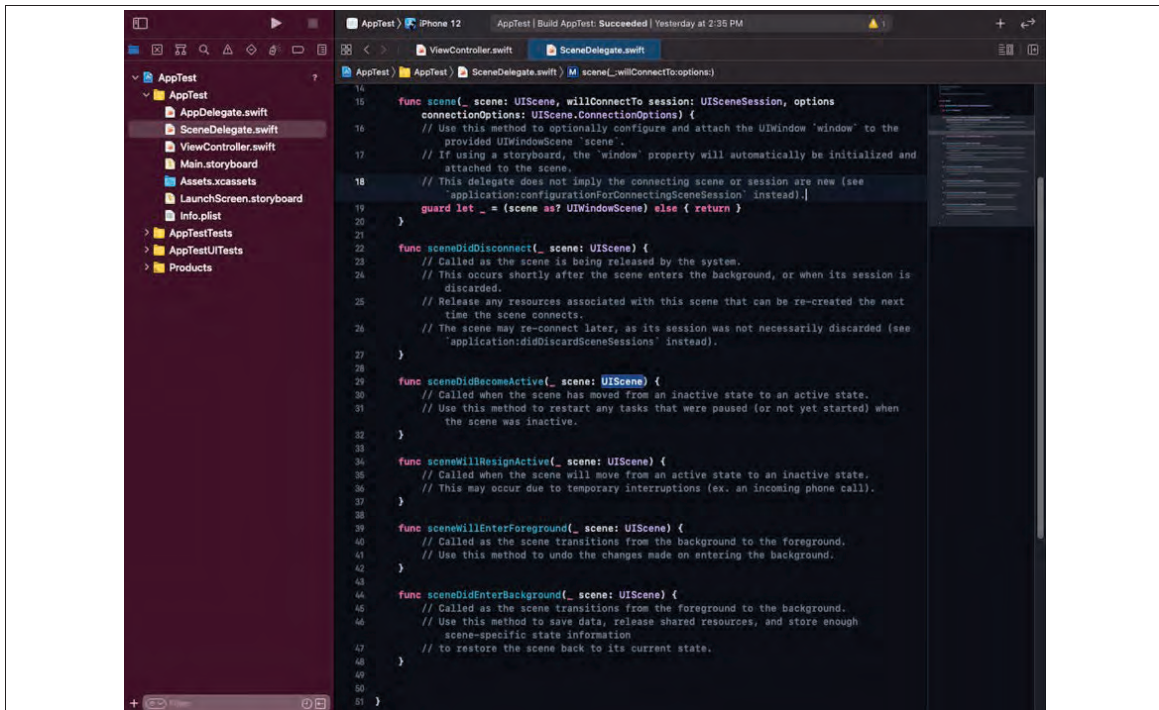


Installing Apple's Xcode Developer Tools results in the installation of infringing Accused Instrumentalities, including Xcode IDE, Instruments, and Simulator.

### Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)

[1pre] A system for testing an application for a mobile device comprising:

An example of an Xcode IDE window from Apple's Xcode Developer Tools for reviewing source code files is shown below:



The above image shows an example window from Xcode IDE. The left pane displays different files which make up an application named "AppTest" (the application being tested). The right pane displays the source code for a file named "SceneDelegate.swift" (which contains source code related to certain visual elements of the sample application). This window is used to review source code, which is often part of software testing, where various tests are run as part of the app development process.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

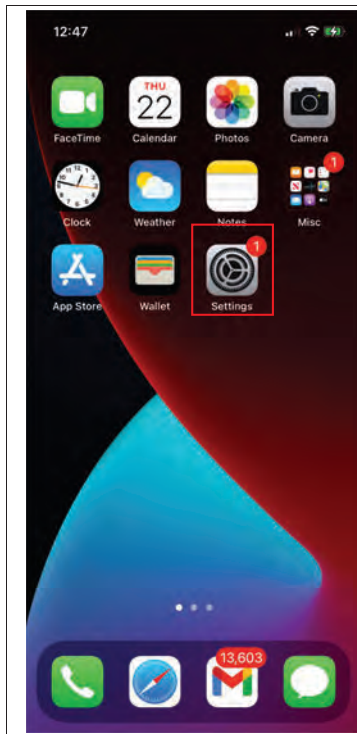
Network Link Conditioner is an application that simulates/emulates network characteristics such as bandwidth limitations, packet loss, latency, etc. Network Link Conditioner may be (1) installed/activated in the settings menu on an iOS device (such as an iPhone), for example, when the developer plugs in that iOS device to a computer running Xcode, or (2) installed on the computer running Xcode by selecting menu options from Xcode. Running Network Link Conditioner allows simulation/emulation of network characteristics.

**1. Network Link Conditioner on iOS Device (e.g., iPhone 12)**

The below screenshots demonstrate how a developer accesses Network Link Conditioner on an Apple iPhone 12 iOS Device. Note that the references to an iPhone 12 (and all other specific device models referenced throughout this document) are illustrative, and infringement is not limited to any specific model or version of iOS or MacOS device.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

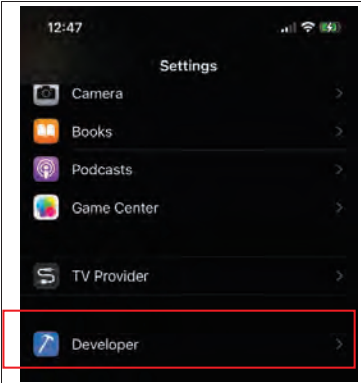
[1pre] A system for testing an application for a mobile device comprising:



This image shows the screen of an Apple iPhone 12 iOS Device. Network Link Conditioner is added under “Settings,” which the user reaches by selecting the icon in the red box annotation.

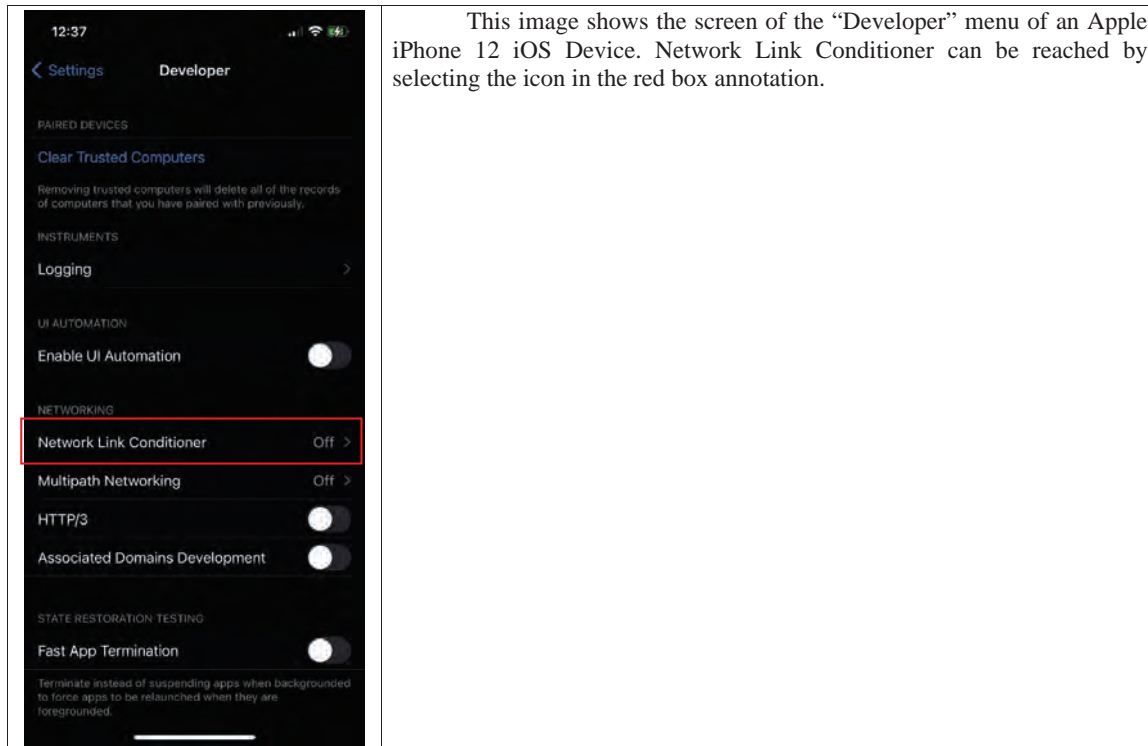
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

 A screenshot of an iPhone 12's Settings application. The screen is dark-themed and shows a list of settings categories: Camera, Books, Podcasts, Game Center, TV Provider, and Developer. The 'Developer' option at the bottom is highlighted with a red rectangular box. The status bar at the top shows the time as 12:47 and signal strength indicators.	<p>This image shows the screen of the Settings menu of an Apple iPhone 12 iOS Device. Network Link Conditioner can be found under the “Developer” option, which the user reaches by selecting the icon in the red box annotation.</p>
---	---

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

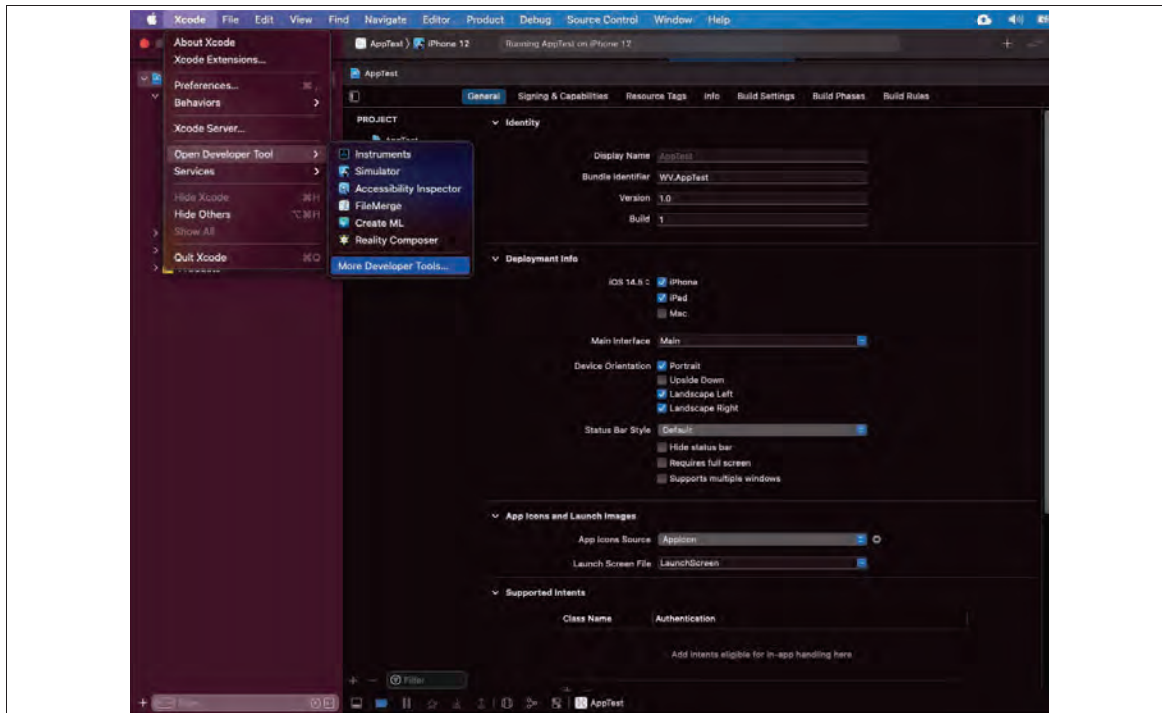


**2. Network Link Conditioner on MacOS Computer/Xcode**

Network Link Conditioner can also be installed on a computer running Apple's MacOS.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:



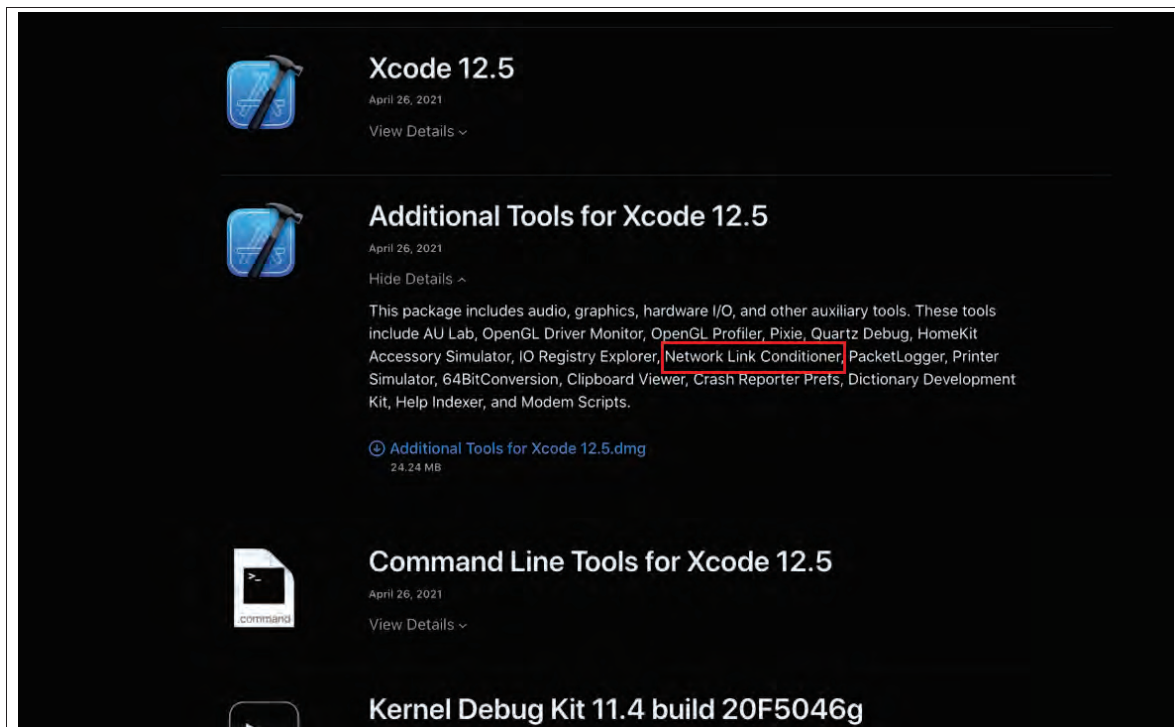
The above image shows the Xcode IDE. A user can access additional Developer Tools by selecting Xcode->Open Developer Tool->More Developer Tools...

Selecting "More Developer Tools..." opens a web browser to a web page where Additional Tools (including Network Link Conditioner) can be downloaded:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:



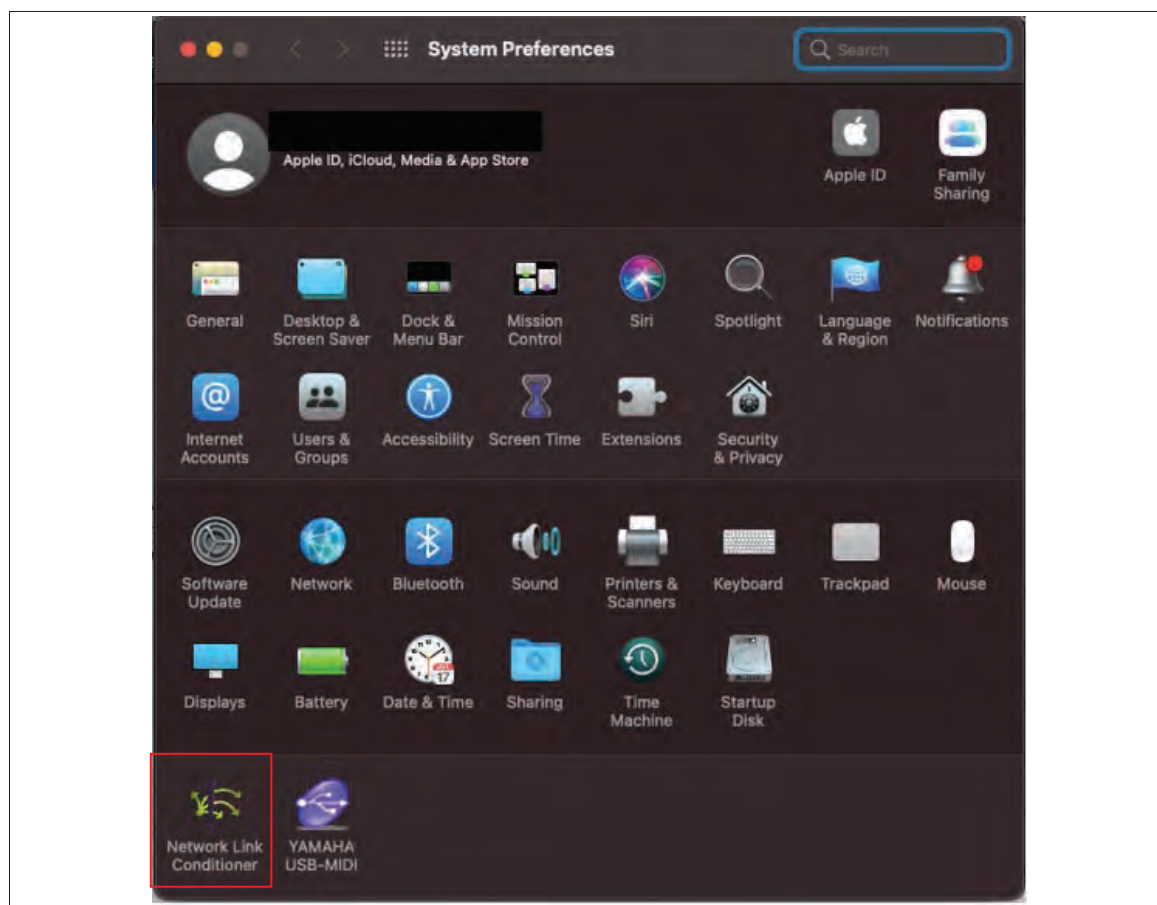
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

Network Link Conditioner is installed to (and executed from) the System Preferences preference pane in MacOS:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1pre] A system for testing an application for a mobile device comprising:

The above image shows the System Preference pane on a MacOS computer. The Network Link Conditioner can be reached by selecting the icon in the annotated red box.

On information and belief, Defendant uses Apple's Xcode Developer Tools to test mobile banking applications for its business, for example—Chase Mobile. While Chase Mobile is identified as an example application, the contentions detailed in this chart apply to all mobile application testing done by or on behalf of Defendant using the Accused Instrumentalities. On information and belief, Defendant's testing of mobile banking applications includes using the features detailed throughout this document.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

**[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;**

As discussed above in Section [1pre] (and incorporated by reference here), the Accused Instrumentalities include a software testing interface. That software testing interface is configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application.

Discussion of this limitation is broken into two sections: (1) provides an overview of how Xcode Developer Tools simulate/emulate characteristics of networks; and (2) provides an overview of how Xcode Developer Tools display resources available to or utilized by applications.

**1. Xcode Developer Tools Emulation/Simulation Overview**

When an Xcode user writes an application, the user can choose to execute that application on either a physical device, or on a simulator by selecting the destination from within Xcode IDE:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

## Select a Simulated Device

For iOS, tvOS, and watchOS apps, you can choose a simulated device, under [Platform] Simulators, from the *run destination menu* next to the scheme menu in the toolbar.

Annotations:

- Choose a real device.
- Choose a generic device to build using platform SDKs.
- Choose simulators of a device family.
- Customize simulators.

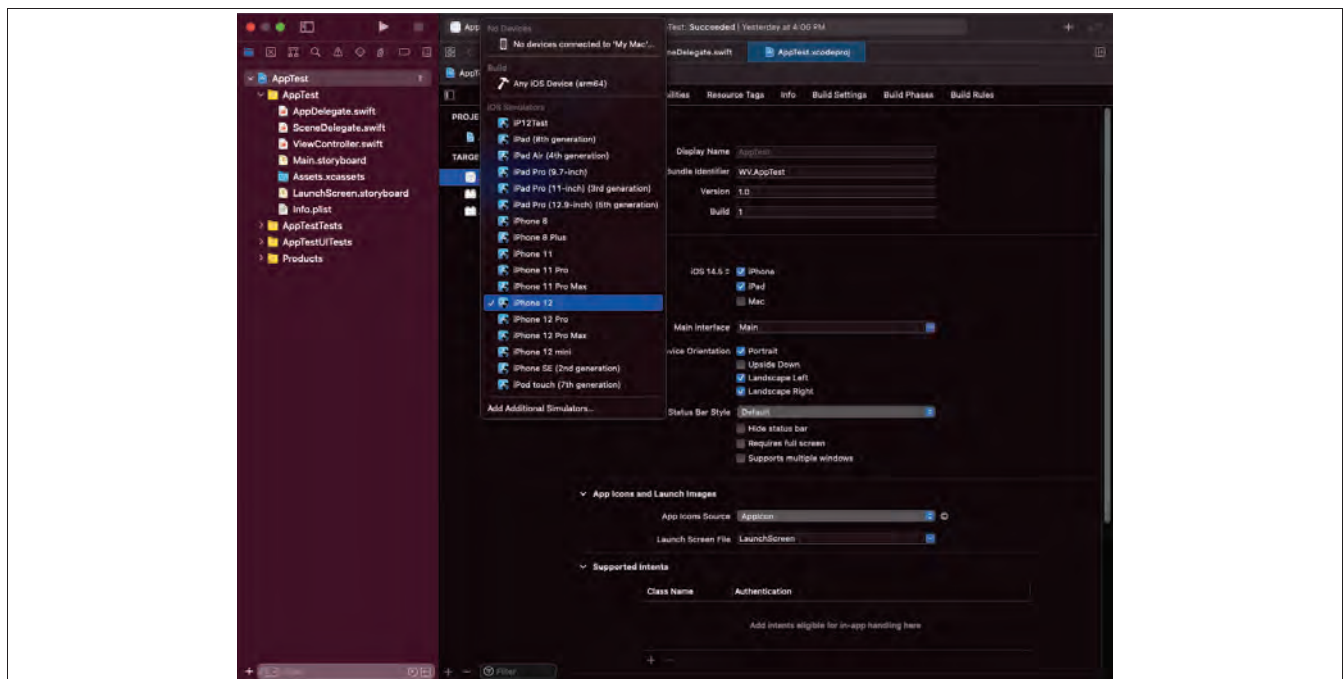
To add additional simulators of a product family running older versions of the operating system, choose Add Additional Simulators.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

<https://developer.apple.com/documentation/xcode/running-your-app-in-the-simulator-or-on-a-device>, accessed on December 28, 2021.

Xcode displays a list of iOS mobile device models (such as iPhones and iPads) from which a user can select to simulate how the application will run on that model. The application being tested can then be deployed on a model that is specific to that device.

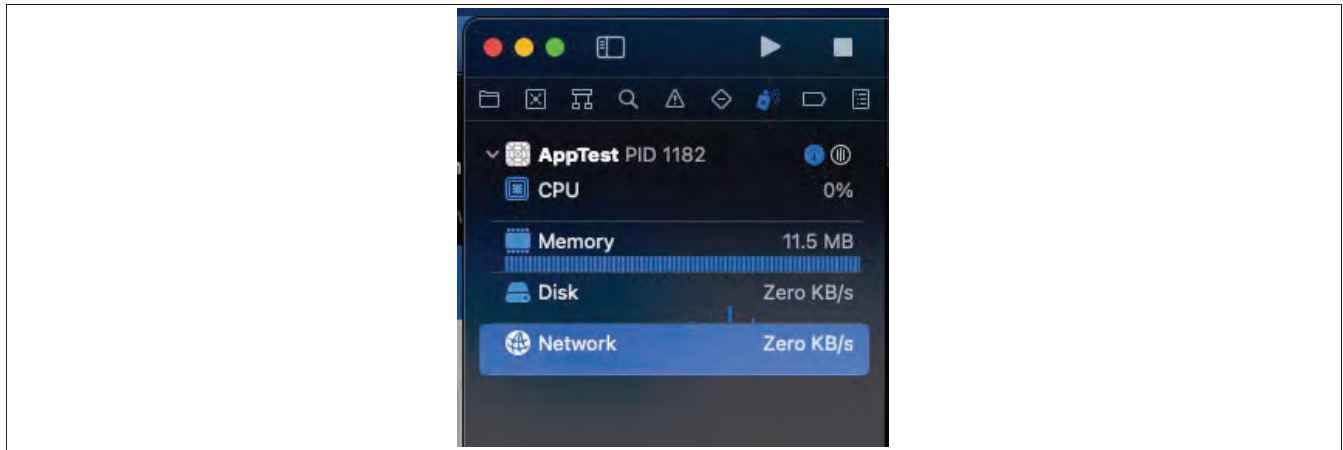


**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Each model includes characteristics indicative of a corresponding mobile device, such as screen size, resolution, and button availability.

Xcode Simulator can also be used to simulate other mobile device characteristics, such as the CPU, Memory, and Disk Storage that would be available to the application if it were running on a physical device. For example, the screenshot below shows resource usage and availability of an application called "AppTest" as it is being run on a simulated iPhone 12:





**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

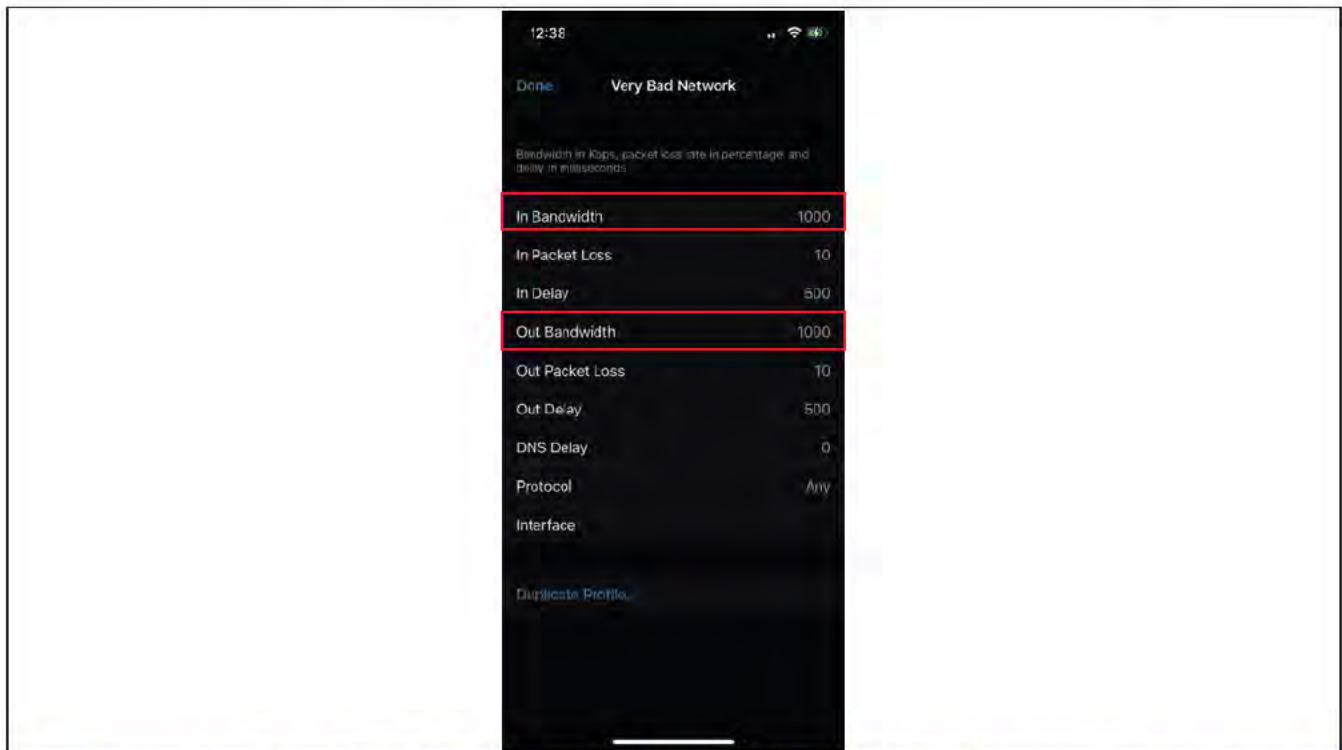
These characteristics are shown for both simulated and physical devices.

Network Link Conditioner (in combination with either Xcode Simulator or a physical device) can be used to emulate/simulate network characteristics indicative of performance of the mobile device when executing the application.

Emulated network characteristics include, *inter alia*, in/out bandwidth, in/out packet loss, in/out delay, and DNS delay. These network characteristics can be set in Network Link Conditioner either on a physical iOS device, or in the Network Link Conditioner in MacOS. For example, by setting bandwidth limits through Network Link Conditioner, a developer can limit the bandwidth available to be used by the application being executed (and thus the network characteristics include at least bandwidth availability).

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



As shown below, Network Link Conditioner settings can be set on both (i) a physical iOS device, as well as (ii) a MacOS device (such as a computer). Setting the Network Link Conditioner settings on a MacOS device simulates/emulates the network characteristics available to be used by applications running on Xcode Simulator on that same MacOS device.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

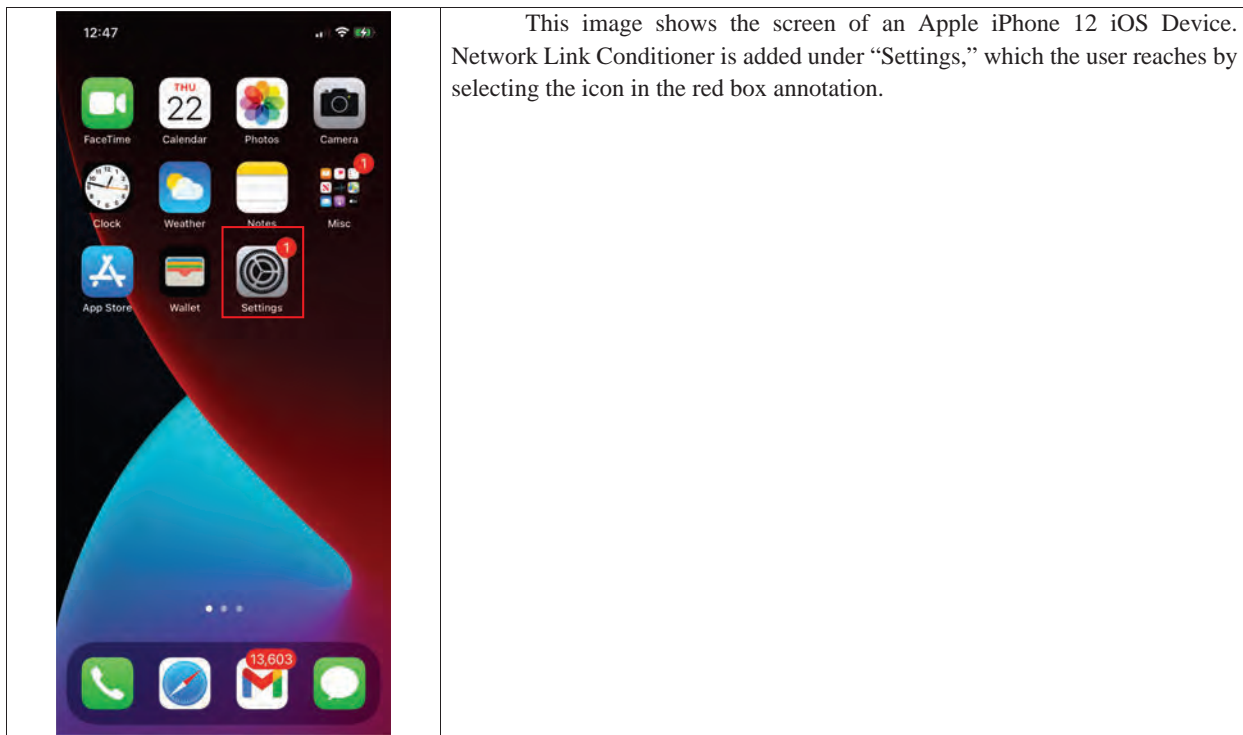
By setting the emulated/simulated network values in Network Link Conditioner, a developer can limit, for example, maximum available bandwidth into or out of the device. After setting these limits, a developer will know what bandwidth is available to the device, for example by comparing current bandwidth usage against the maximum bandwidth available. These characteristics are characteristics of an operator network, such as a 3G network operated by an operator.

*a. Network Link Conditioner on iOS*

The following steps are performed on an iOS Device, such as an iPhone 12. Note that the use of a specific model or version device is merely exemplary, and not meant to be limiting.

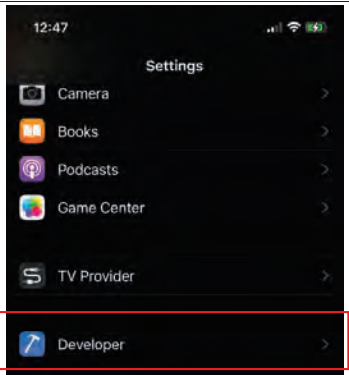
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



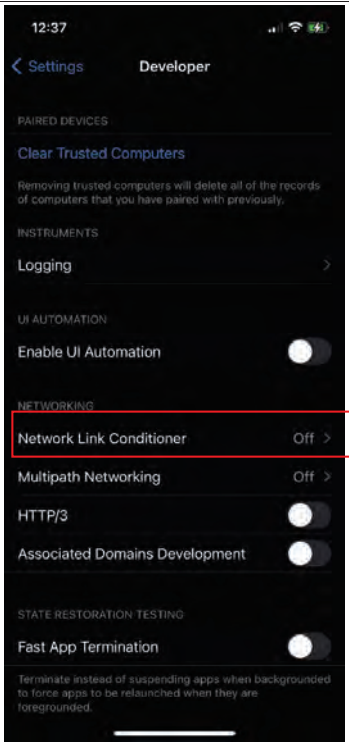
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

	<p>This image shows the screen of the Settings menu of an Apple iPhone 12 iOS Device. Network Link Conditioner can be found under the “Developer” option, which the user reaches by selecting the icon in the red box annotation.</p>
--	---

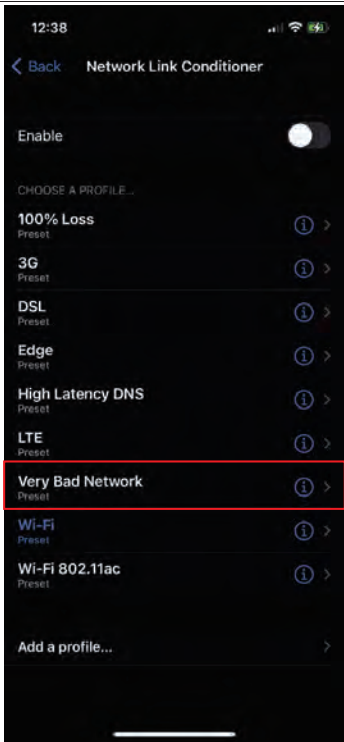
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

	<p>This image shows the screen of the “Developer” menu of an Apple iPhone 12 iOS Device. Network Link Conditioner can be reached by selecting the icon in the red box annotation.</p>
--	---

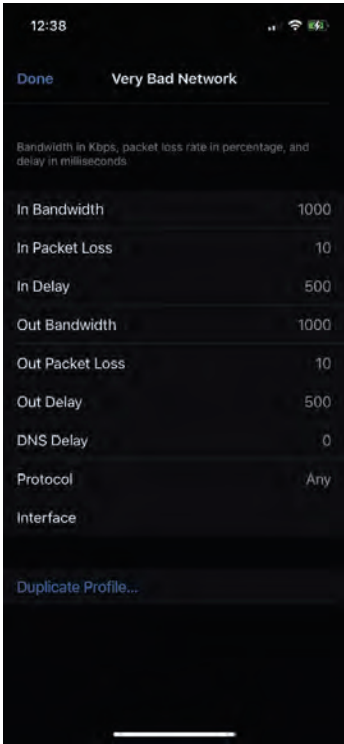
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

	<p>This image shows the Network Link Conditioner menu. The toggle button at the top of the menu indicates whether Network Link Conditioner is enabled on the device. The user can select a Network Profile which determines which network characteristics will be emulated/simulated. For example, if the user selects the “Very Bad Network,” then the user will be taken to the screen below.</p>
--	---

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

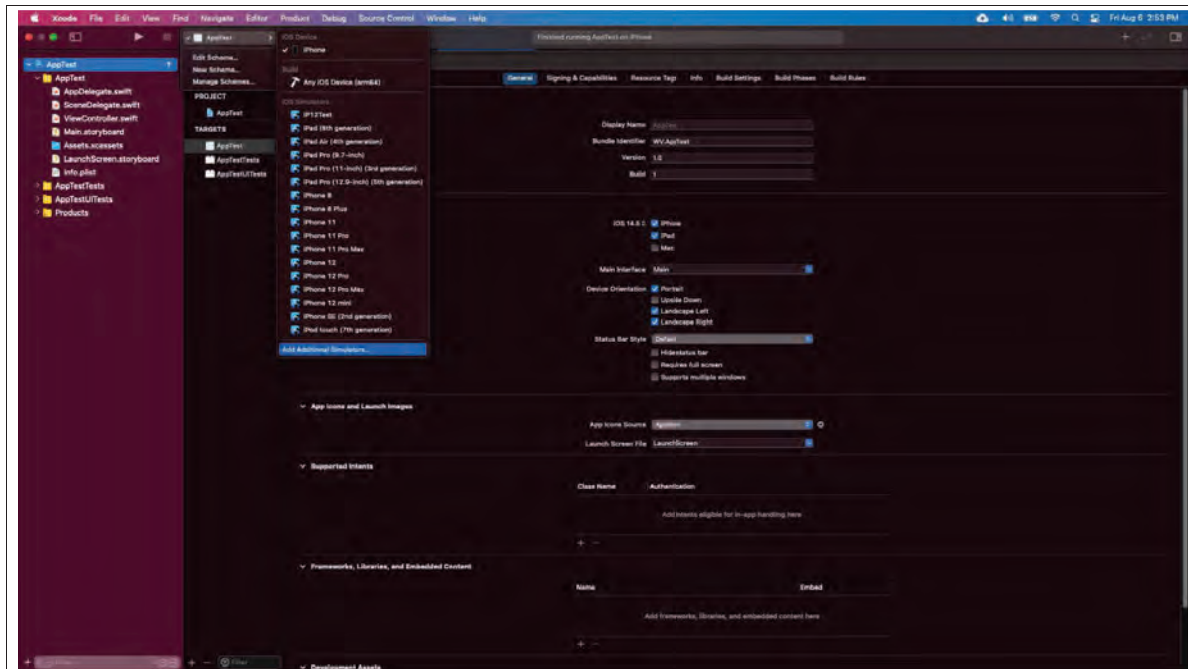
 <p>12:38</p> <p>Done Very Bad Network</p> <p>Bandwidth in Kbps, packet loss rate in percentage, and delay in milliseconds</p> <table border="1"> <tbody> <tr> <td>In Bandwidth</td> <td>1000</td> </tr> <tr> <td>In Packet Loss</td> <td>10</td> </tr> <tr> <td>In Delay</td> <td>500</td> </tr> <tr> <td>Out Bandwidth</td> <td>1000</td> </tr> <tr> <td>Out Packet Loss</td> <td>10</td> </tr> <tr> <td>Out Delay</td> <td>500</td> </tr> <tr> <td>DNS Delay</td> <td>0</td> </tr> <tr> <td>Protocol</td> <td>Any</td> </tr> <tr> <td>Interface</td> <td></td> </tr> </tbody> </table> <p>Duplicate Profile...</p>	In Bandwidth	1000	In Packet Loss	10	In Delay	500	Out Bandwidth	1000	Out Packet Loss	10	Out Delay	500	DNS Delay	0	Protocol	Any	Interface		<p>This image shows the characteristics of the “Very Bad Network” selected by the user, including in/out bandwidth, in/out packet loss, in/out delay. After setting these limits, a developer will know what bandwidth is available to the device, for example by comparing current bandwidth usage against the maximum bandwidth available.</p>
In Bandwidth	1000																		
In Packet Loss	10																		
In Delay	500																		
Out Bandwidth	1000																		
Out Packet Loss	10																		
Out Delay	500																		
DNS Delay	0																		
Protocol	Any																		
Interface																			



### Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Link Conditioner settings can also be set for the device directly through Xcode IDE. While these settings are made through Xcode IDE, they will be effective on the iOS device connected to the MacOS computer:

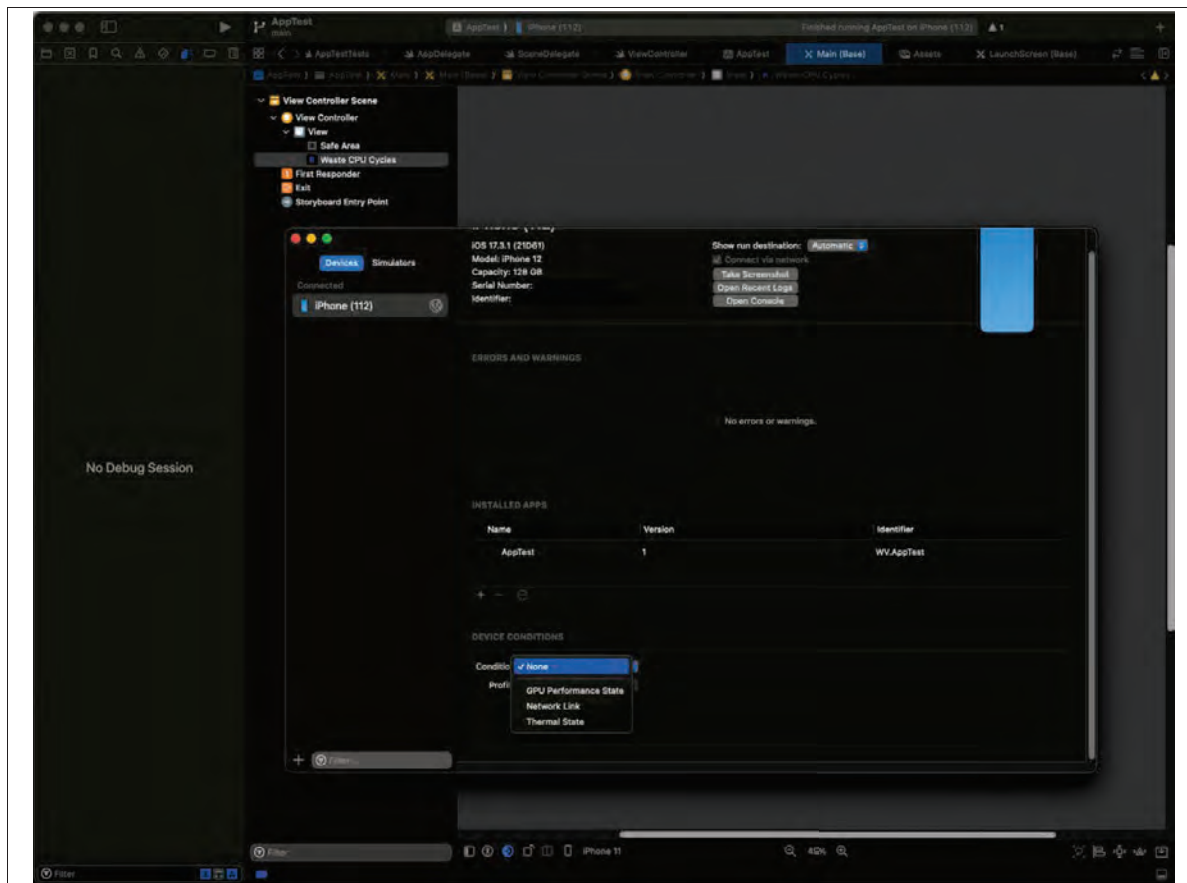


Users can access the Link Conditioner settings from Xcode IDE by selecting the “Add Additional Simulators...” menu setting.

By selecting the “Devices” option, users can set Device conditions.

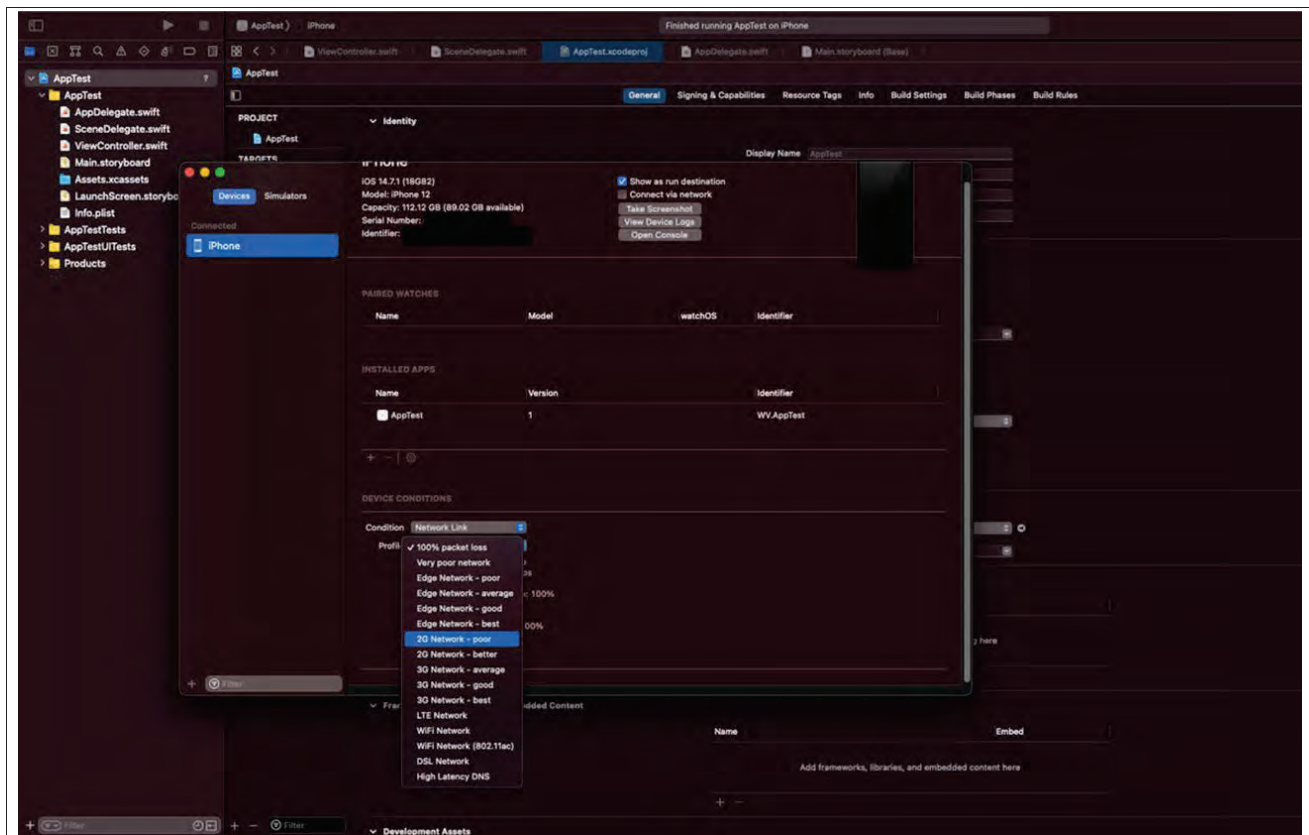
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



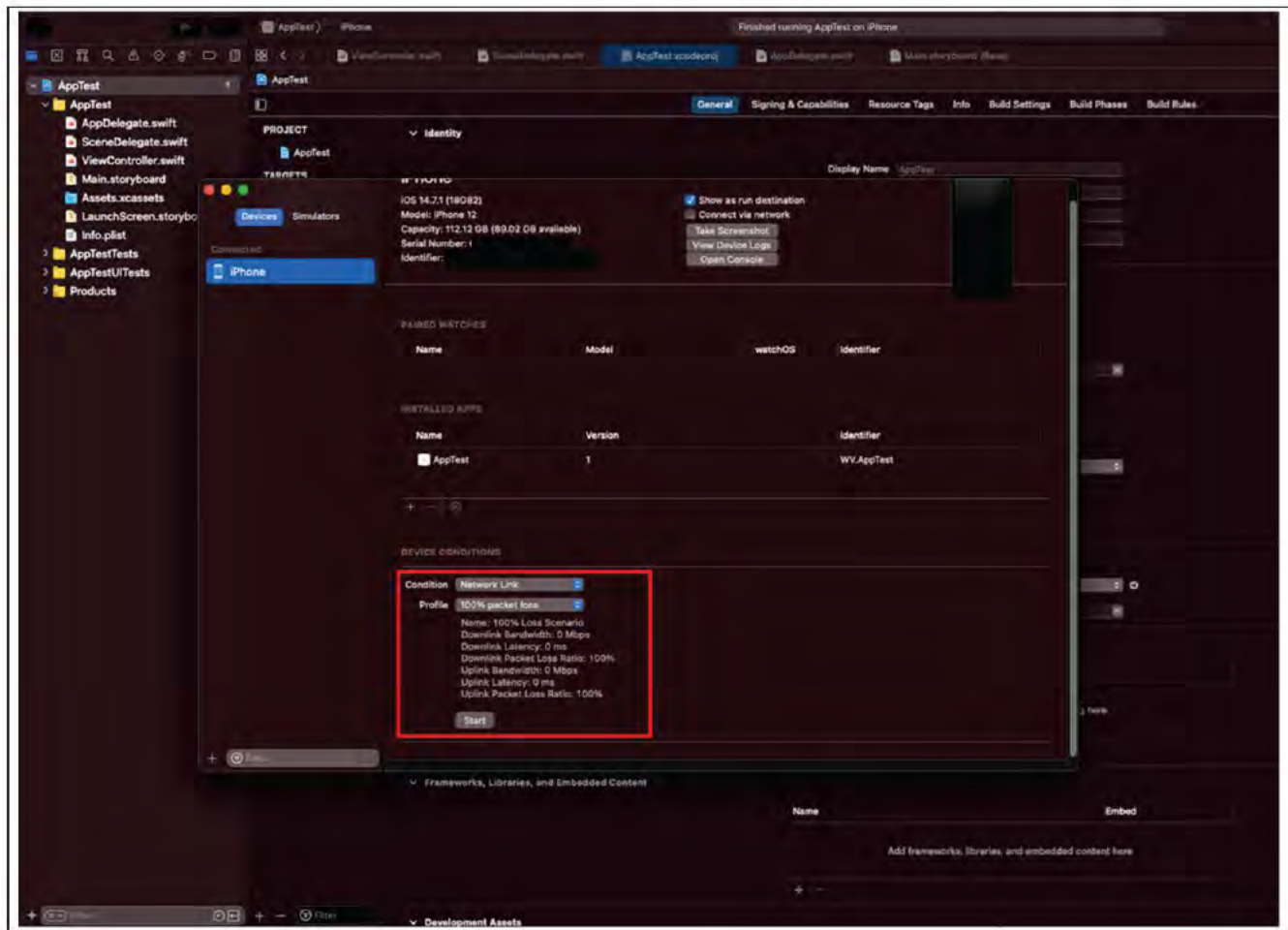
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

If users select "Network Link" in the first dropdown box, the user can then select a profile representing network characteristics in the second dropdown box.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

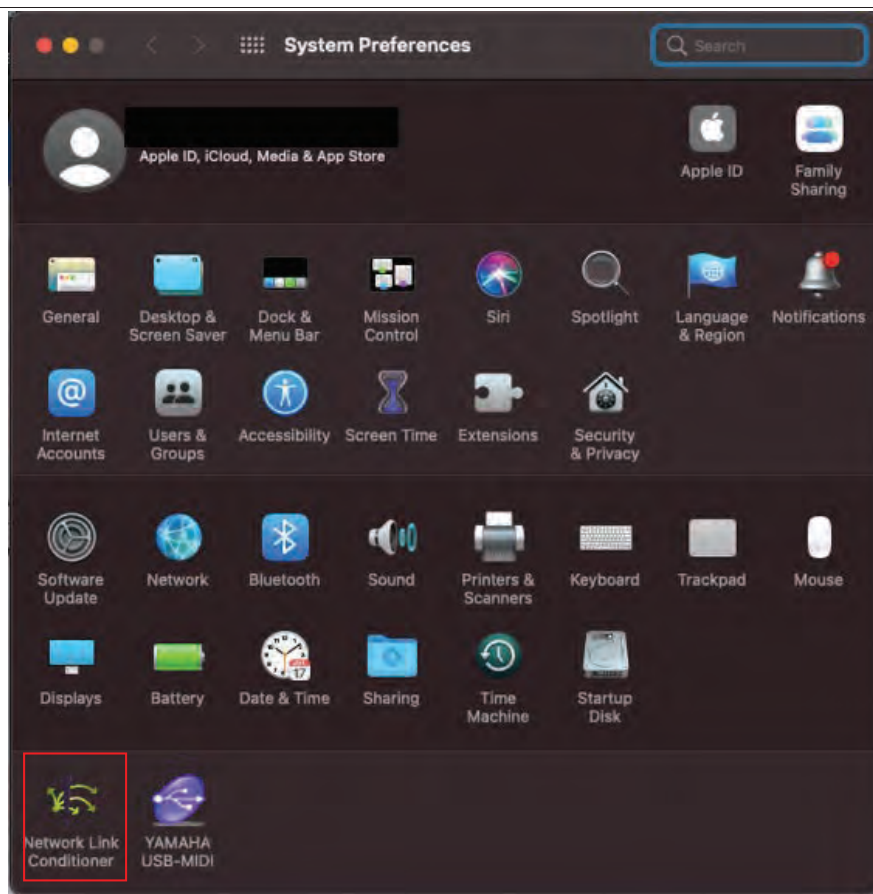
As shown above, after selecting the “100% packet loss” profile, the characteristics (including bandwidth, latency, and packet loss) are selected based on that profile.

*b. Network Link Conditioner on MacOS*

Users can also emulate/simulate network characteristics indicative of performance of the mobile device when executing the application by using the Network Link Conditioner tool from System Preferences in MacOS.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Users select the Network Link Conditioner (shown in red) from System Preferences in MacOS.



From Network Link Conditioner, users can select a profile (such as “100% Loss”, “3G”, etc.) which simulates/emulates network characteristics.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



Simulated network characteristics include bandwidth, packets dropped, and delay. After setting these values, a developer can determine the resources available to the application by comparing the current resource utilization (such as bandwidth) against the values set in Network Link Conditioner.

**2. Xcode Developer Tool Display Overview**

Apple's Xcode Developer Tools are used to visually display the resources available to, and used by, the application. These resources are displayed through tools such as (a) Xcode IDE and (b) Xcode Instruments.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

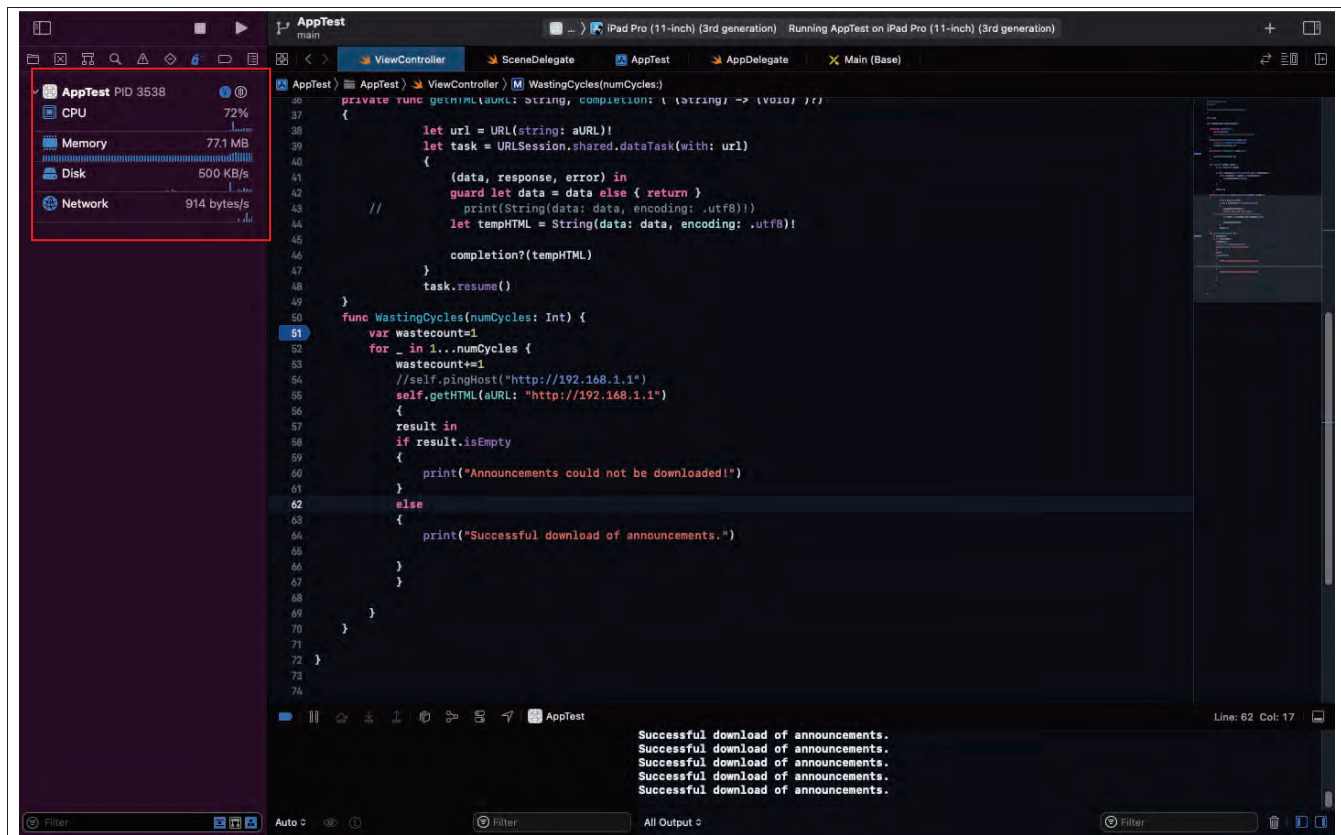
[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

*a. Xcode IDE*

When a developer runs the application being tested through Xcode IDE, then Xcode IDE displays a plurality of visual representations of the resources available to (and being used by) the application, with each bar in the bar graph being displayed at the same time and indicating the resources being used and that are available:

### Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



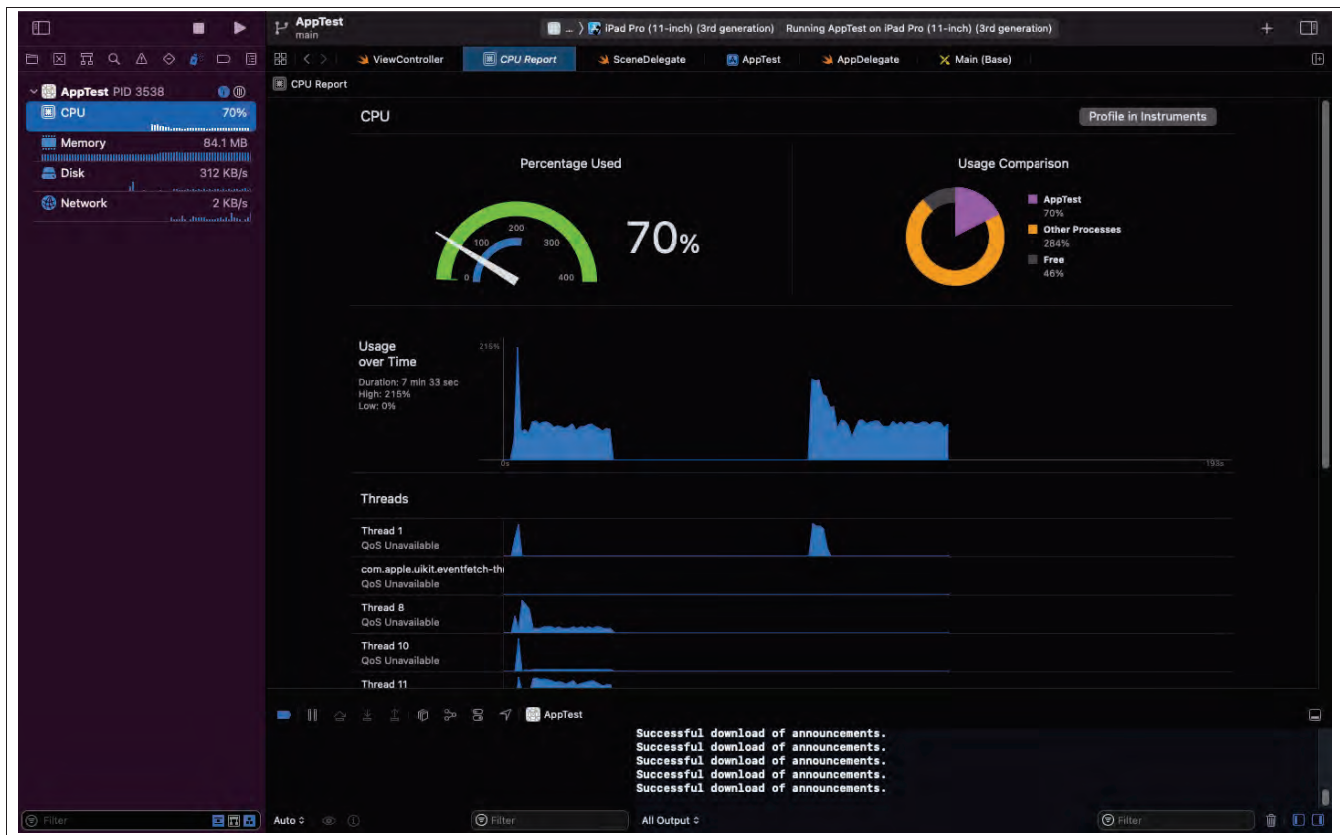
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Developers can view additional visual representations of the resources used and available by clicking on each individual graph. For example, clicking CPU shows the following (including the percentage currently used (and the percentage available, e.g., 100% minus the percentage being used), usage over time, usage by thread, and a comparison of the application's CPU usage vs other processes):

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



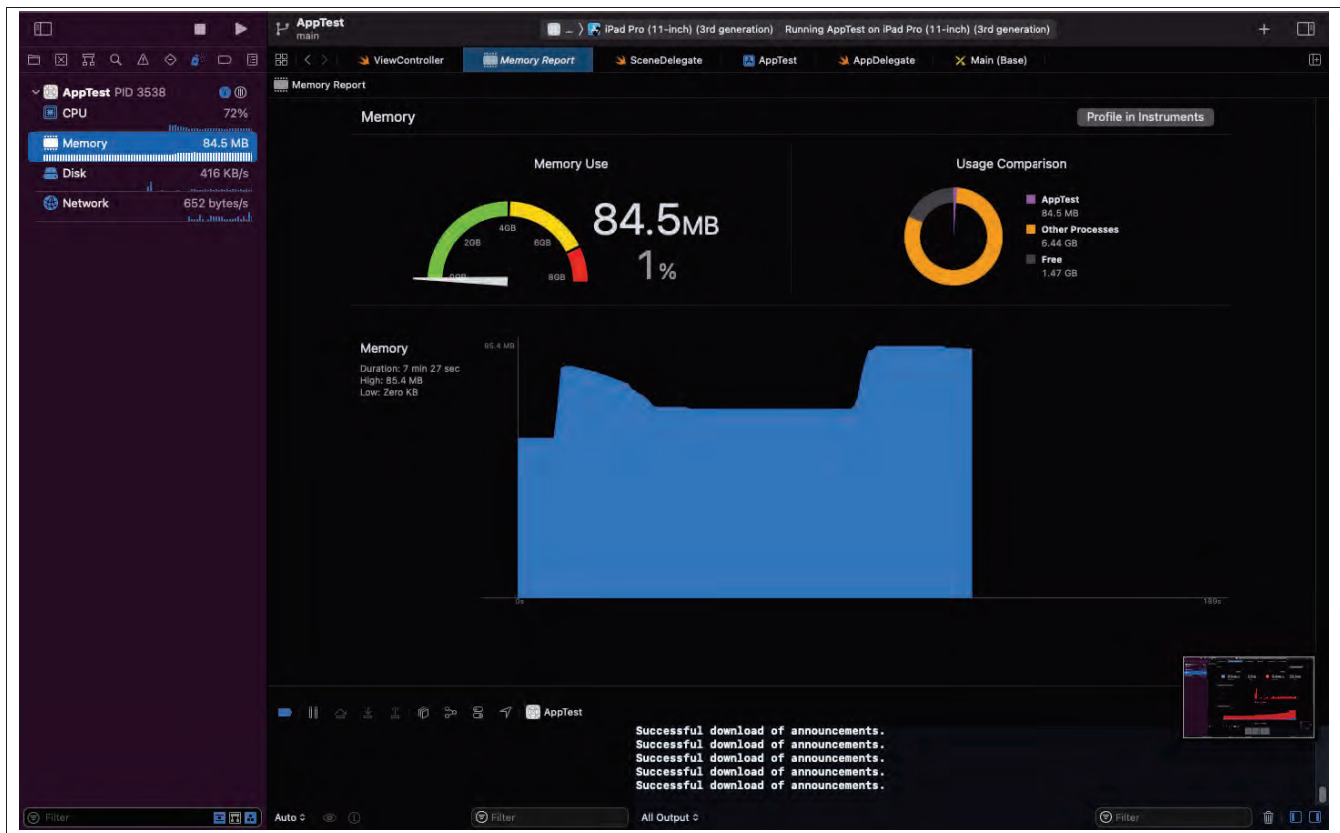
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Similarly, clicking Memory provides a visual display including graphs showing memory currently used, total memory available, % currently used, usage comparison of the application vs other processes, and memory usage over time. Furthermore, developers can compare the memory currently used against the memory available in the physical device:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Disk Usage (showing current bytes read/written per second, total bytes read or written, reading and writing rates over time, and total reads and writes over time) Furthermore, developers can compare the disk characteristics currently used against the disk characteristics in the physical device:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

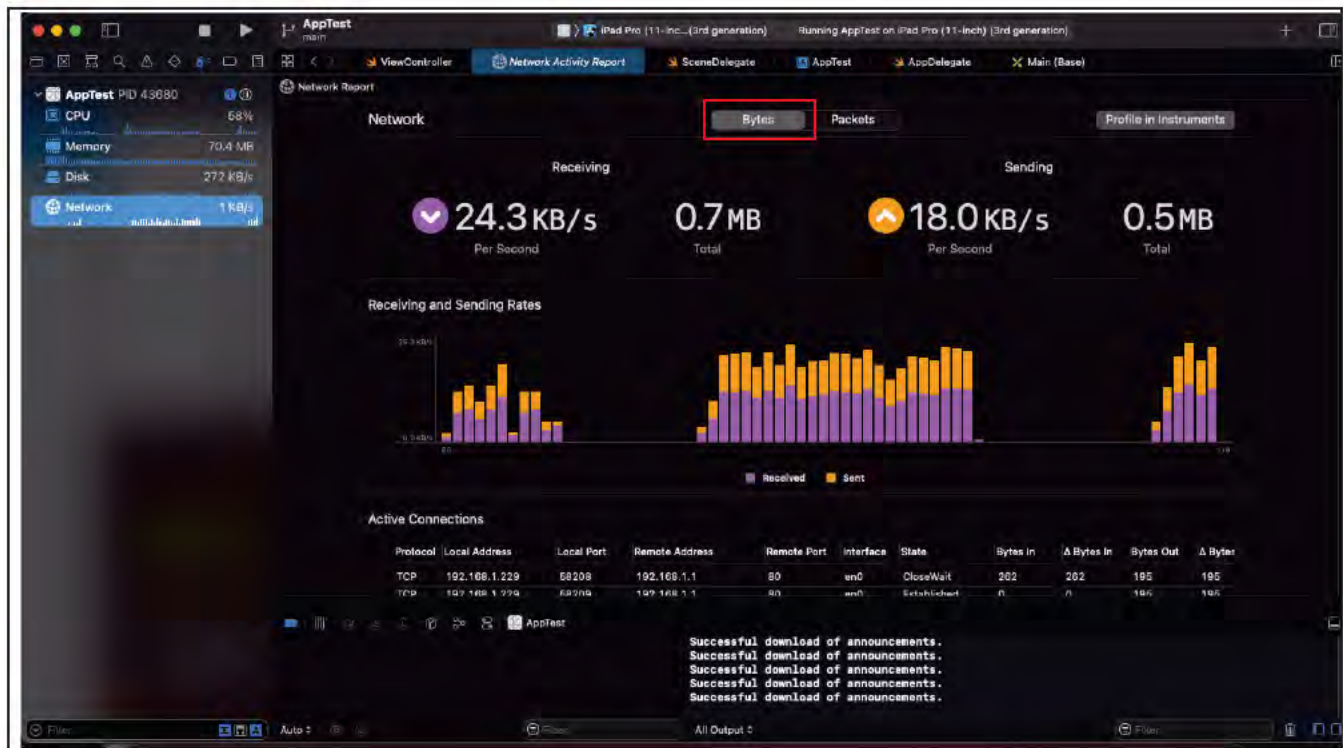
The screen above shows current disk usage, as well as the disk usage resources still available. Furthermore, a developer could use known maximum disk read/write rates to determine available resources based on utilized resources.

The images below show Network Activity. Network Activity is shown in terms of either Bytes or Packets, depending on which option is selected. A developer could use the limitations selected in Network Link Conditioner (such as maximum bandwidth rates) to determine available resources based on utilized resources.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

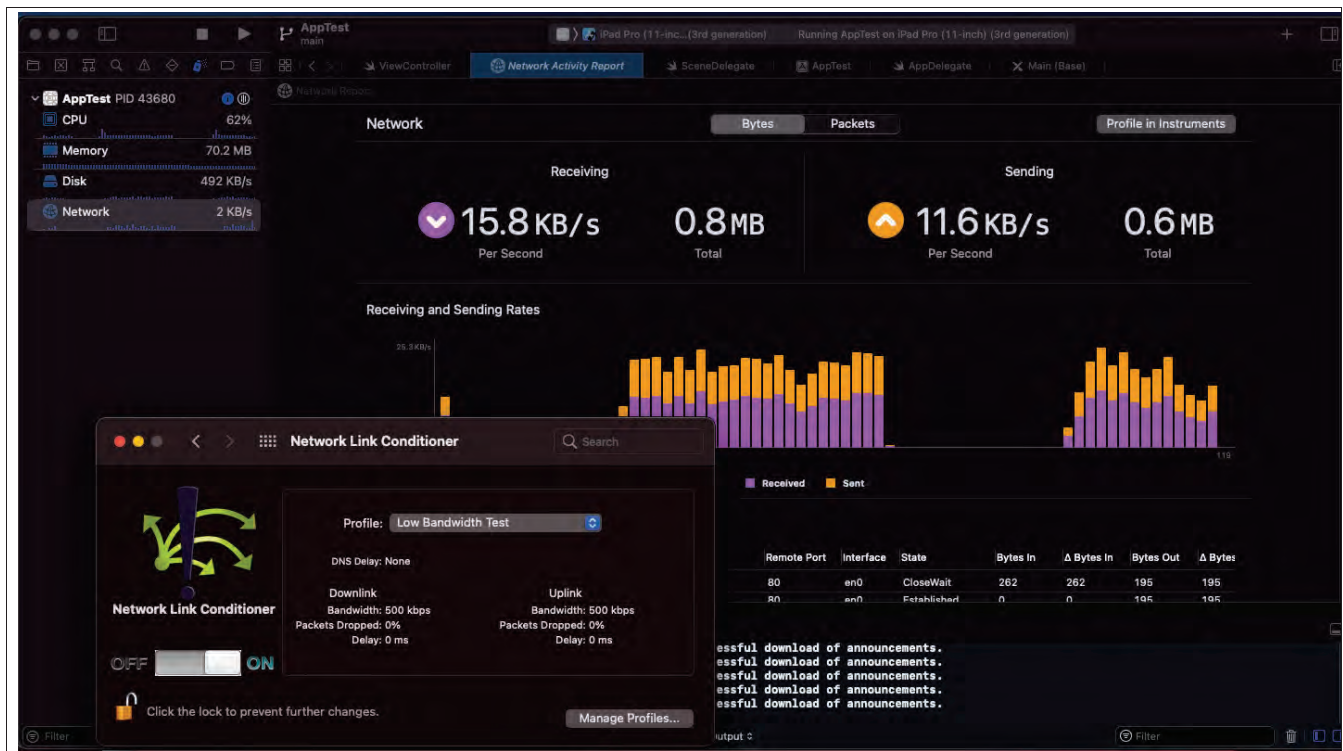
[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

The Bytes option is shown below (displaying current sending and receiving rates, total data sent and received, and sending and receiving rate speed over time):



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



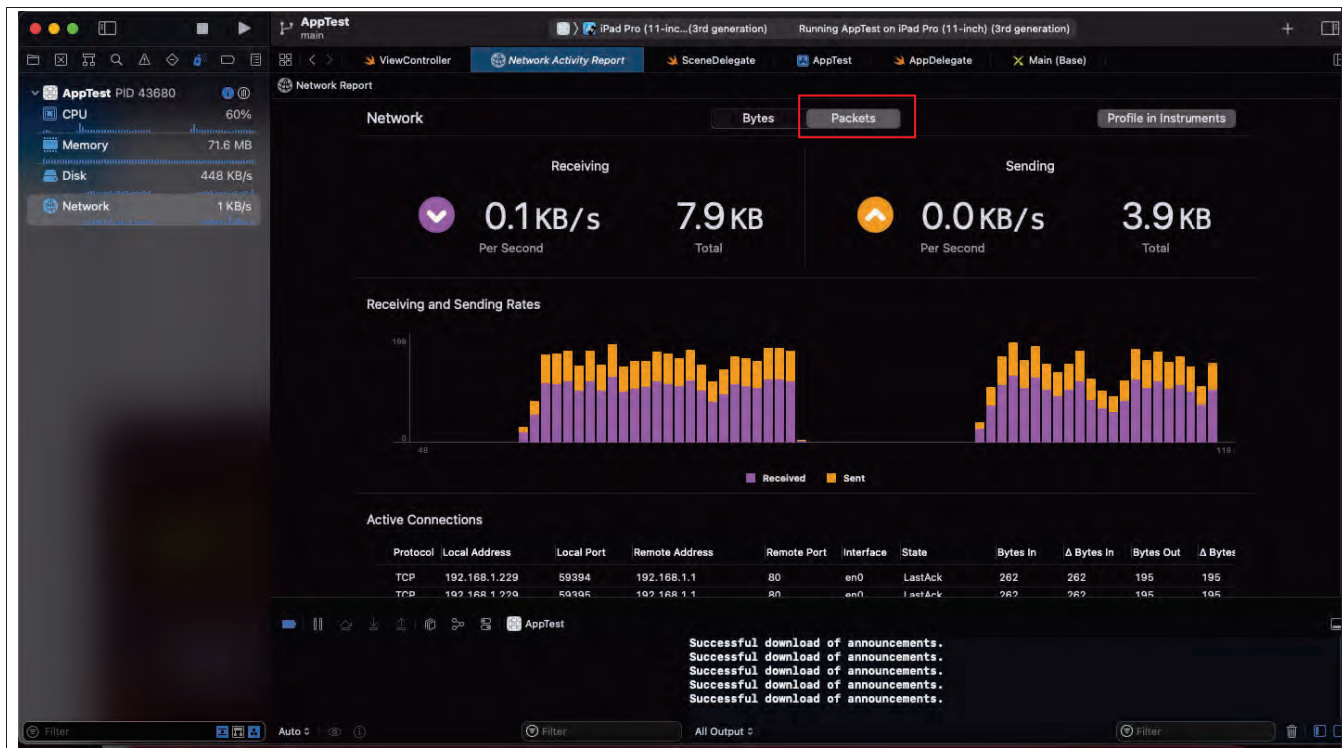
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

The Packets option is shown below (displaying current sending and receiving rates, total data sent and received, and sending and number of packets sent or received over time). The timing and quantity of packets sent correspond to the network latency and packet loss.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

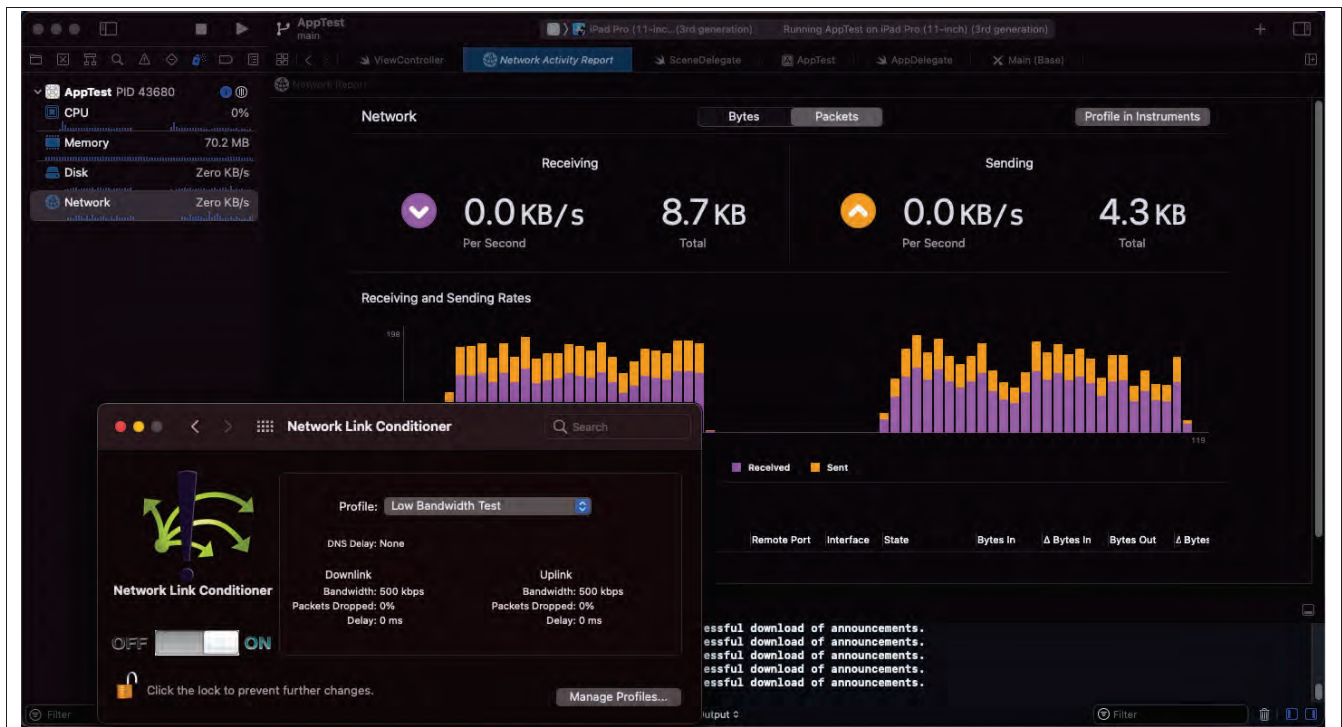
[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;





**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

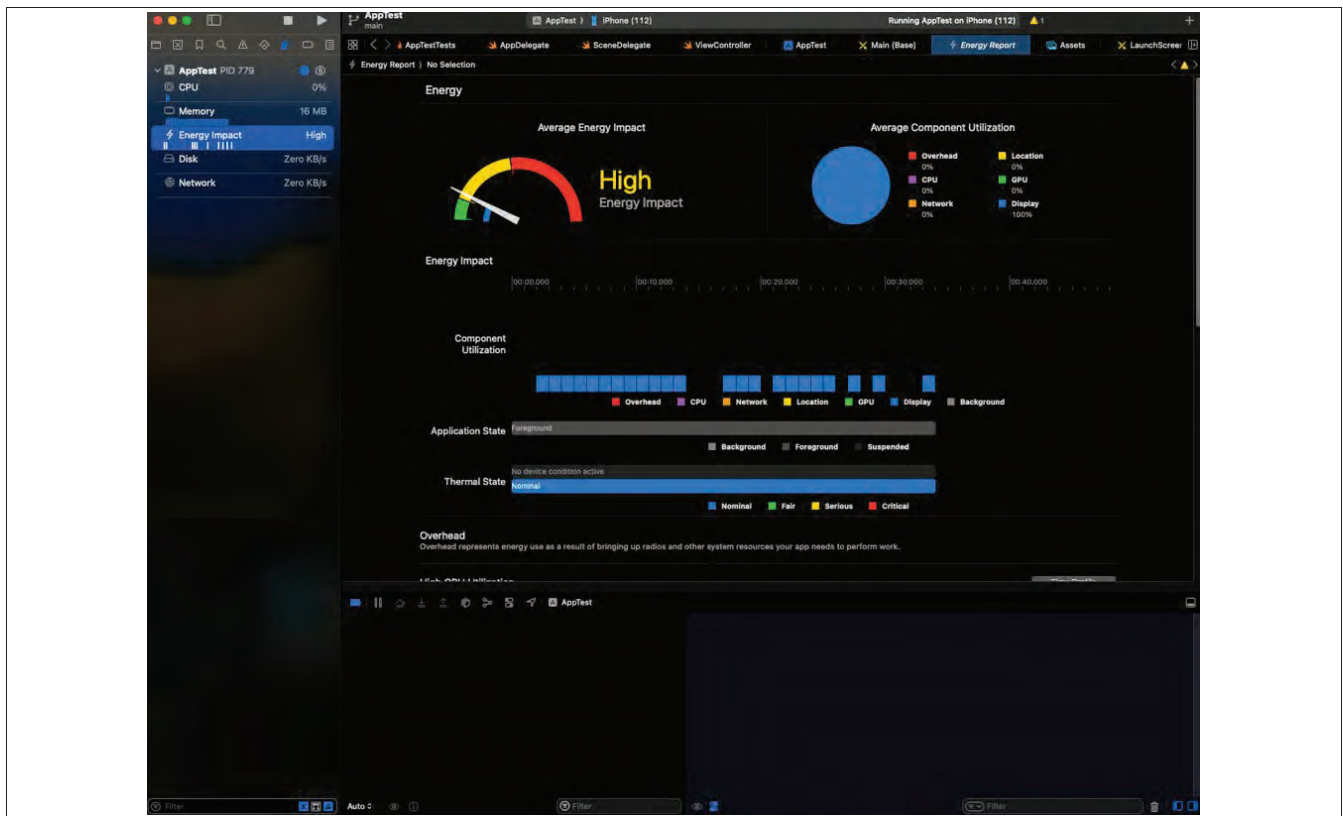
[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



### Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

The Energy Impact option shown below shows current energy impact of the application, as well as energy still available.





**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

*b. Xcode Instruments*

The Xcode Instruments tool is also used to visually display the resources available to, and used by, the application. As described by Apple, the “Instruments” tool:

Instruments is a powerful and flexible performance-analysis and testing tool that's part of the Xcode tool set. It's designed to help you profile your iOS, watchOS, tvOS, and macOS apps, processes, and devices in order to better understand and optimize their behavior and performance. Incorporating Instruments into your workflow from the beginning of the app development process can save you time later by helping you find issues early in the development cycle.

In Instruments, you use specialized tools, known as *instruments*, to trace different aspects of your apps, processes, and devices over time. Instruments collects data as it profiles, and presents the results to you in detail for analysis.

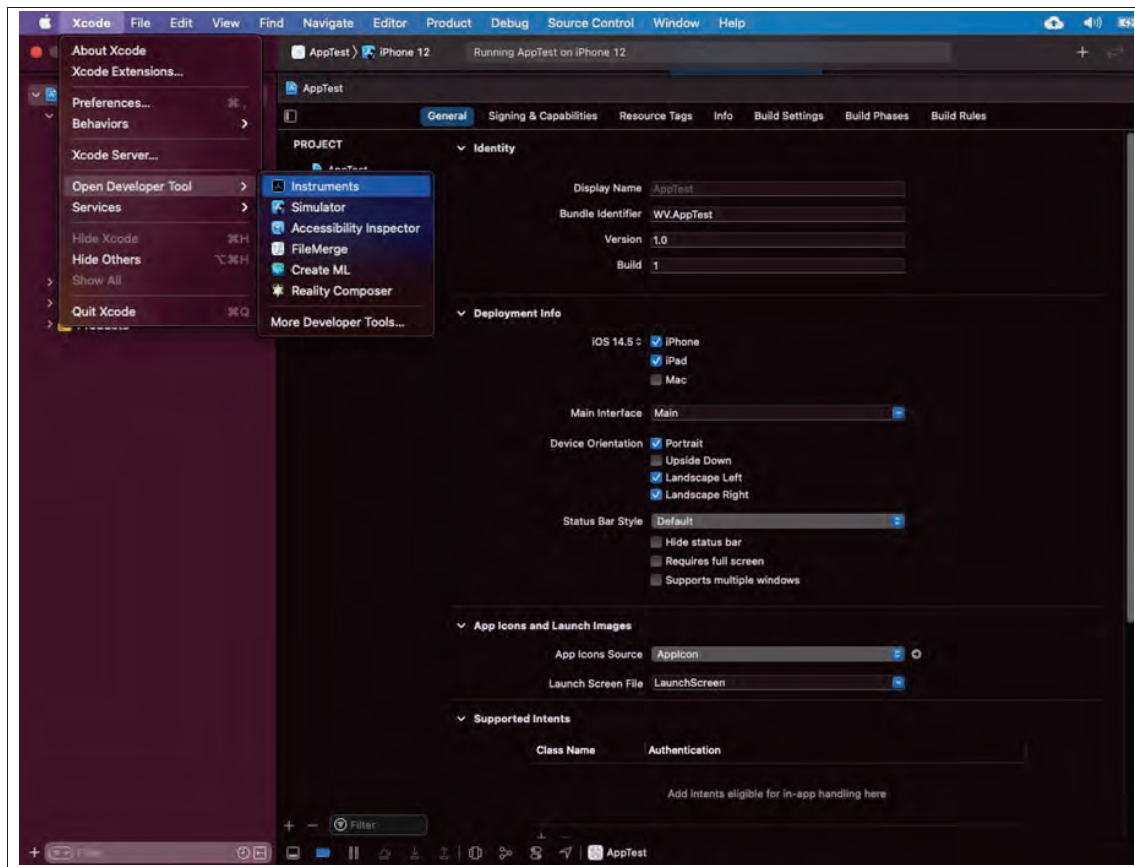
Unlike other performance and debugging tools, Instruments allows you to gather widely disparate types of data and view them side by side. This makes it easier to identify trends that might otherwise be overlooked. For example, your app may exhibit large memory growth caused by multiple open network connections. By using the Allocations and Connections instruments together, you can identify connections that are not closing and thus resulting in rapid memory growth.

<https://help.apple.com/instruments/mac/current/#/dev7b09c84f5>, accessed on March 12, 2024.

To access Xcode Instruments, an Xcode IDE user may select the “Instruments” menu option from Xcode’s menu:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

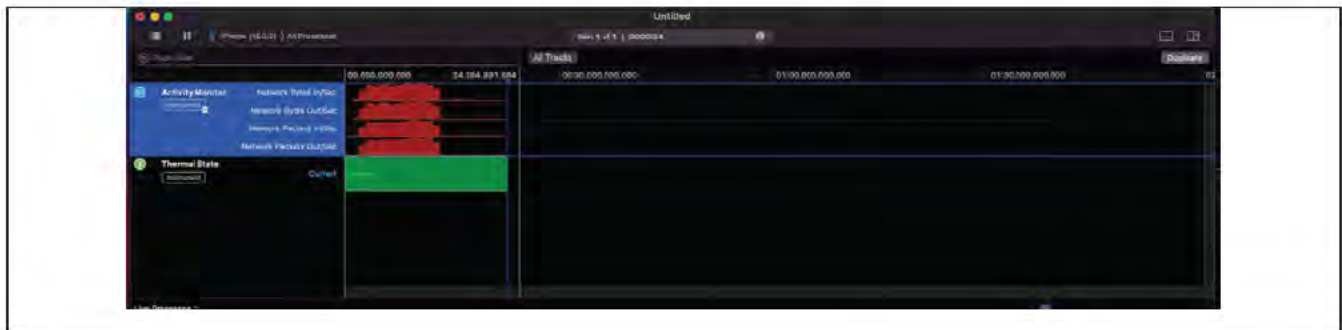
Multiple Instrument templates are available related to various resources, as shown in the examples below:



The Instruments window displays one or more windows showing simultaneous displays of resources of the mobile device that are used by and/or available to the application currently running:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

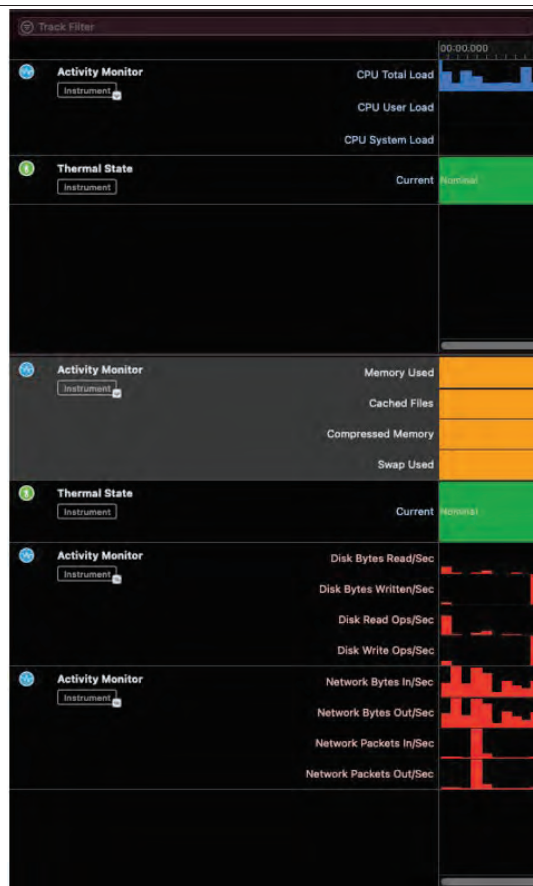
[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



Resources displayed in instruments include those related to CPU (Total Load, User Load, System Load), Thermal State, Memory (Memory Used/Available (for example, the total memory minus the memory used), Cached Files, Compressed Memory, Swap Used), Disk (Bytes Read or Written/Second, Disk Read or Write Operations/Second), Network interactions (Network Bytes In or Out/Second, Network Packets In or Out/Second), GPU Usage, and others:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1a] a software testing interface configured to simultaneously visually simulate, via one or more profile display windows, a plurality of operator network characteristics including at least bandwidth availability indicative of performance of the mobile device when executing the application;

Xcode Instruments users may select what they wish to see, and multiple graphs may be displayed at once. The size of the bar in each bar graph shows both the resources utilized, as well as the resources available to the application.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.

**[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.**

The network characteristics simulated by Xcode Network Link Conditioner (including bandwidth availability) are based at least in part on data predetermined from interactions between one or more mobile devices and at least one operator network.





**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.

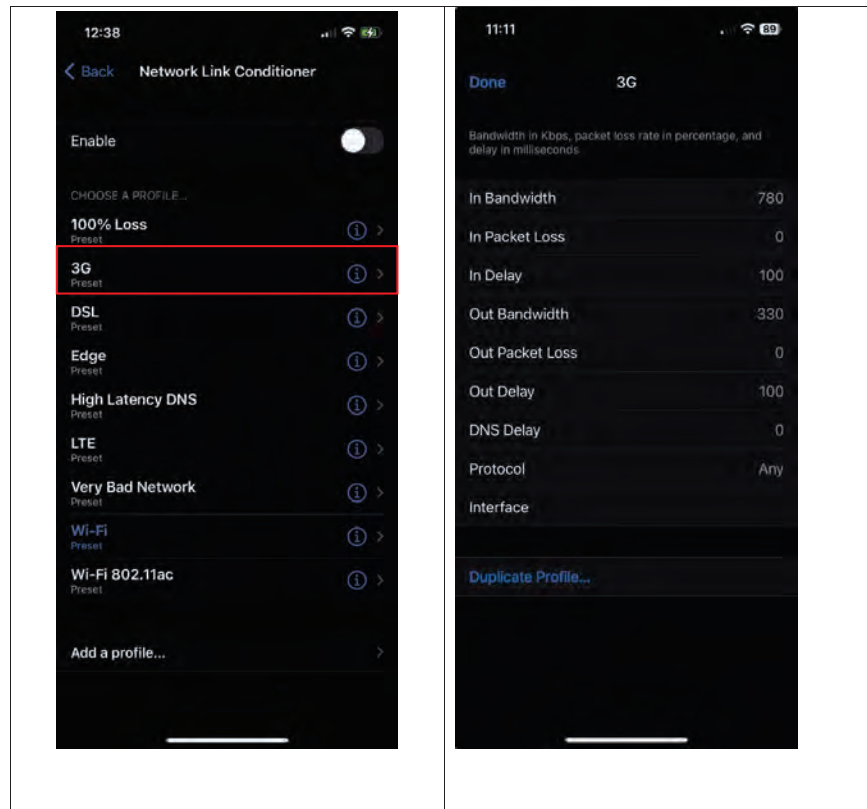


Network Link Conditioner on MacOS.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

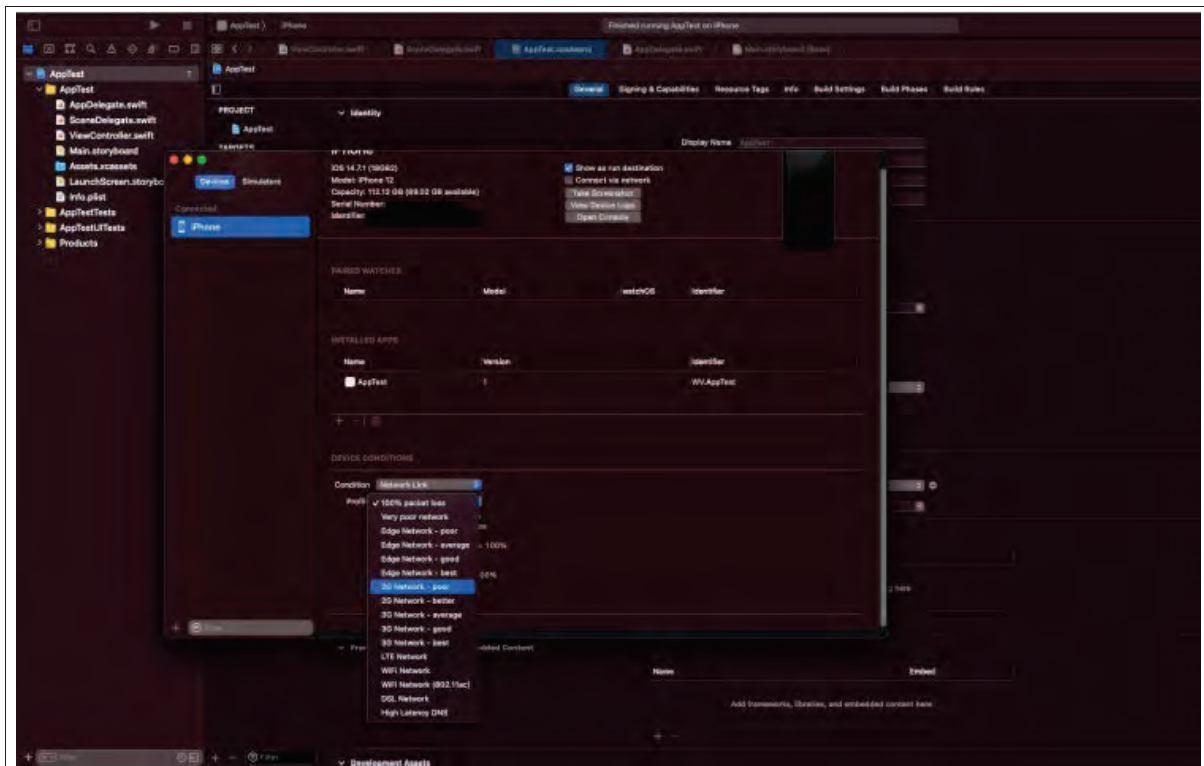
[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.



Network Link Conditioner on iOS.

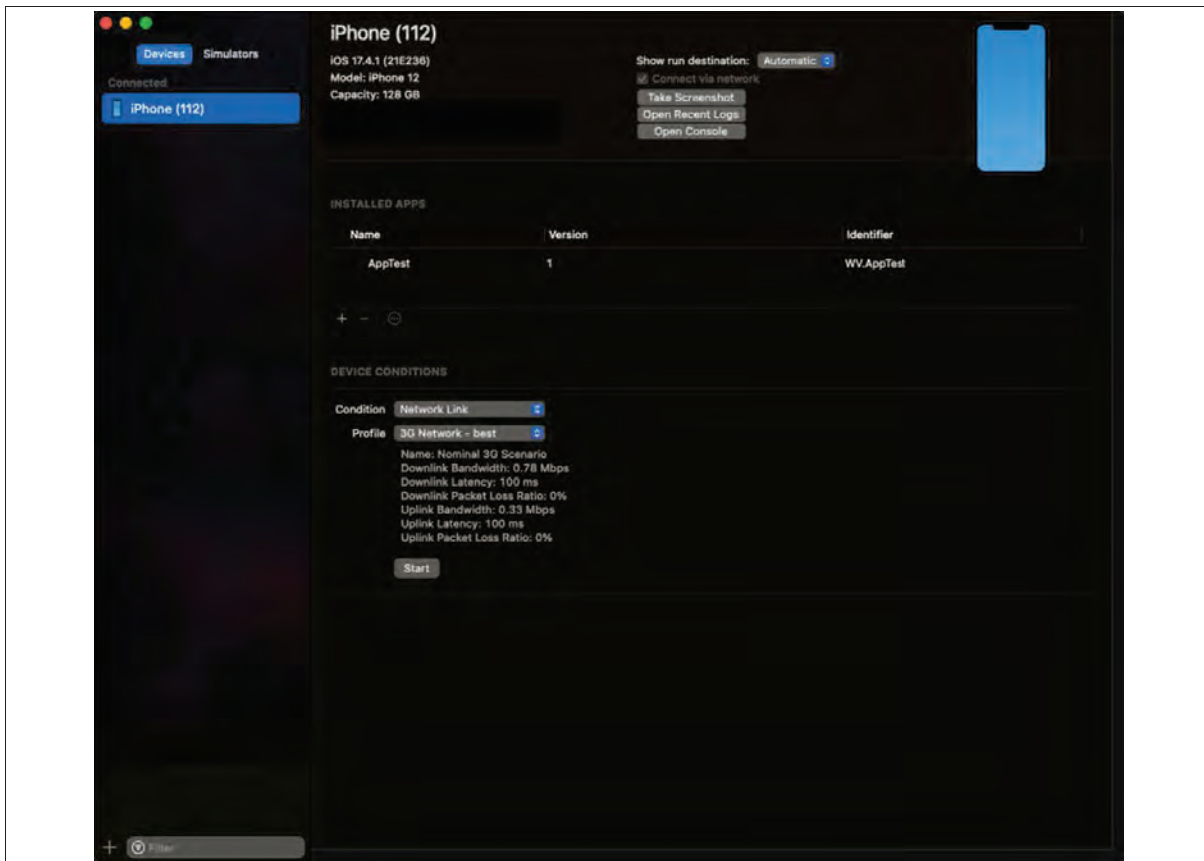
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.



Network Link Conditioner on iOS configured through Xcode IDE.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.

The network characteristics such as bandwidth are based on data of interactions between one or more mobile devices and at least one operator network at least because of their reliance on network standards developed to dictate the operation of real-world networks, such as LTE, 2G, 3G, and EDGE.

For instance, GSM corresponds to what is often referred to as the “2G” network standard established around 1991. Similarly, EDGE corresponds to Enhanced Data rates for GSM Evolution, an enhancement to GSM. And HSCSD corresponds to High Speed Circuit Switched Data, which is an enhancement to the data rate of circuit switched data in a GSM network. GPRS corresponds to a packet-oriented enhancement to 2G networks—General Packet Radio Service. UMTS is the Universal Mobile Telecommunications System, a new architecture that provided the basis for what is often referred to as the “3G” network standards. HSDPA, or High-Speed Downlink Packet Access, is an enhancement to the 3G network architecture to boost data capacity and improve download rates. Finally, LTE, or Long Term Evolution, represents the transition from 3G to what is typically referred to as “4G” network technology.

Each of these network standards defines the general operation of the network, and these definitions provide theoretical constraints on the networks’ capacity for communication, including bandwidth. These constraints can be further impaired based on network conditions, including the presence of physical obstacles, electro-magnetic interference, and/or distance between the base station and a mobile station with which it is communicating.

The development and evolution of these standards relied on data of interactions with real-world implementations of such networks at least for testing and proof-of-concept. Thus, Network Link Conditioner’s bandwidth constraints correspond to each identified standard, which are based on data of interaction of networks in non-simulated environments.

Many network operators have participated in (and continue to participate in) the cellular standards detailed above. This participation includes both influencing the direction of the standards as well as testing the viability or early implementation of proposed standards.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.

**a. The 3GPP Standards Development Process**

Verizon is participating in and influencing the 5G standards setting process through the 3rd Generation Partner Project (3GPP), which has previously provided LTE, LTE-Advanced and LTE Advanced Pro for commercial cellular/mobile systems. There are seven organizational partners in 3GPP which work on the standards and also several peripheral organizations that reference or provide input to 3GPP standards (Figure 1).

<https://www.verizon.com/business/resources/whitepapers/first-principles-for-securing-5g/> (last visited 5/17/2024); *see also id.* (“The standards development process, including work on security features, benefits from input from companies with real-world experience deploying new technology. It is common for companies like Verizon who are “first movers” to deploy service using new technology while the standards are still in development.”).

**AT&T Teams Up with Global Technology Leaders for Faster 5G Deployment**

AT&T\* is working with several global technology leaders and operators to align on 5G. The efforts are in preparation for the release of the official 3GPP specifications which will form the basis of the global standards.

....

In 5G trials, AT&T is accelerating over-the-air interoperability testing based on standards developed under the 3GPP New Radio (NR) specifications. The trials are designed to easily evolve with future versions of the official 5G standards, a milestone 3GPP targets for 2018. Focusing on the NR standards helps ensure the technology will work correctly with any future 3GPP 5G NR updates.

[https://about.att.com/story/faster\\_5g\\_deployment.html](https://about.att.com/story/faster_5g_deployment.html) (last visited 5/17/2024).

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.

Verizon has said that its goal is to collaborate with vendors on some early specifications and then contribute those to the 3GPP, the mobile industry standards body responsible for creating the 5G standard. Verizon insists that its early release of these 5G specifications won't create fragmentation in the industry.

....

When Verizon released its 5G spec last July, the company said the guidelines were primarily for testing and validating 5G components that will help chipset vendors and others develop interoperable 5G gear and assist with pre-standard testing and fabrication.

<https://www.sdxcentral.com/articles/news/verizon-att-spar-5g-standards-process/2016/09/> (last visited 5/17/2024).

When Network Link Conditioner was released, wireless networks had been widely deployed by network operators. On information and belief, Network Link Conditioner's bandwidth, packet loss, and latency/delay constraints are based on data of interaction with such networks in non-simulated environments operated by network operators.

These protocols are built for use in non-simulated environments, notably, operator networks. Therefore, Network Link Conditioner's bandwidth settings are based on data of interactions between one or more mobile devices and at least one operator network.

This is further reinforced by Apple's documentation, which instructs developers to use Network Link Conditioner to simulate "real-world conditions":

**Test Under Various Conditions**

Xcode provides a tool called Network Link Conditioner that can simulate various network conditions, including reduced bandwidth, high latency, DNS delays, packet loss, and so on. Before you ship any software that uses networking, you should install this tool, enable it, then run your software to see how it performs under real-world conditions.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[1b] wherein the bandwidth availability is based at least in part on bandwidth data predetermined from interactions between one or more mobile devices and at least one operator network.

<https://developer.apple.com/library/archive/documentation/NetworkingInternetWeb/Conceptual/NetworkingOverview/WhyNetworkingIsHard/WhyNetworkingIsHard.html>, accessed on March 13, 2024.

The “real-world conditions” described are based on data predetermined from interactions between one or more mobile devices and at least one operator network.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[2] The system of claim 1, wherein the software is configured to enable a user to select from one or more connection simulations for testing how well mobile content performs on the mobile device.

**Claim 2**

**[2] The system of claim 1, wherein the software is configured to enable a user to select from one or more connection simulations for testing how well mobile content performs on the mobile device.**

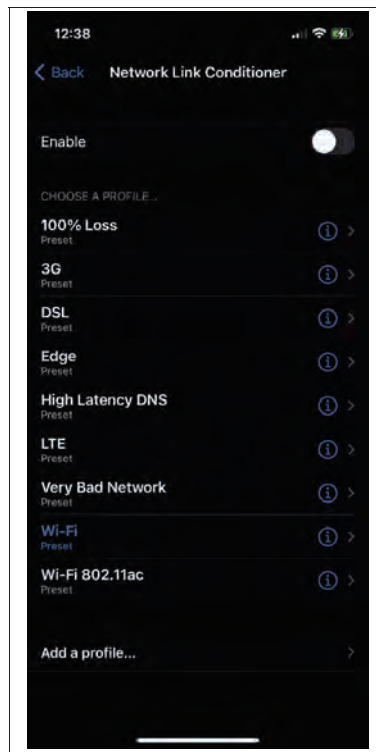
The software including Apple's Xcode Developer Tools is configured to enable a user to select one or more connection simulations for testing how well mobile content performs on the mobile device. These simulations include simulating how the mobile content performs on the mobile device when using, for example, 3G, LTE, and Wi-Fi networks, as discussed above for Claim 1 (and incorporated here by reference). By selecting and running the application in development on a variety of network configurations (such as those listed above), the user can test how well the mobile content (such as the content of the application) performs on the mobile device (such as seeing how the application handles packet loss, limited bandwidth, etc.).



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[2] The system of claim 1, wherein the software is configured to enable a user to select from one or more connection simulations for testing how well mobile content performs on the mobile device.

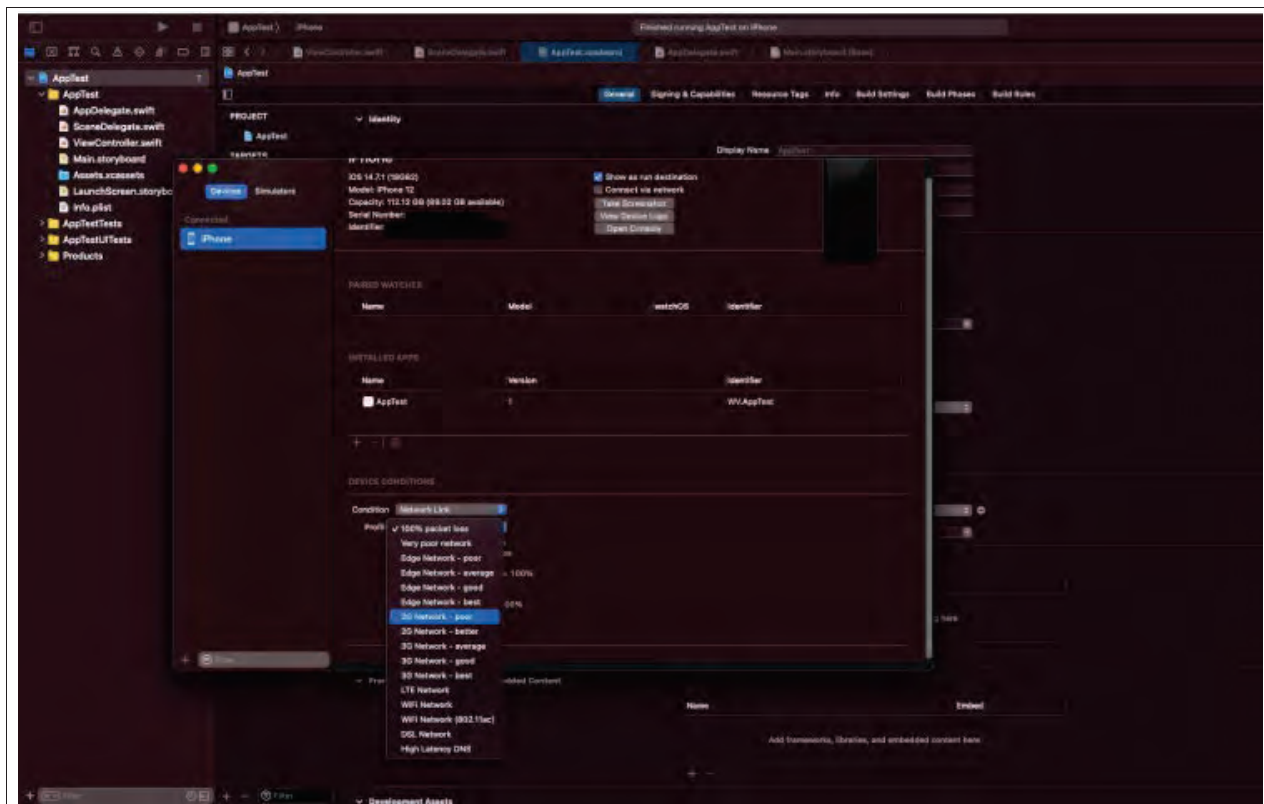
**1. Network Link Conditioner on iOS**



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[2] The system of claim 1, wherein the software is configured to enable a user to select from one or more connection simulations for testing how well mobile content performs on the mobile device.

Network Link Conditioner on iOS configured through Xcode IDE:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[2] The system of claim 1, wherein the software is configured to enable a user to select from one or more connection simulations for testing how well mobile content performs on the mobile device.

**2. Network Link Conditioner on MacOS**



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

- [3] The system of claim 2, wherein the one or more connection simulations are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation.

**Claim 3**

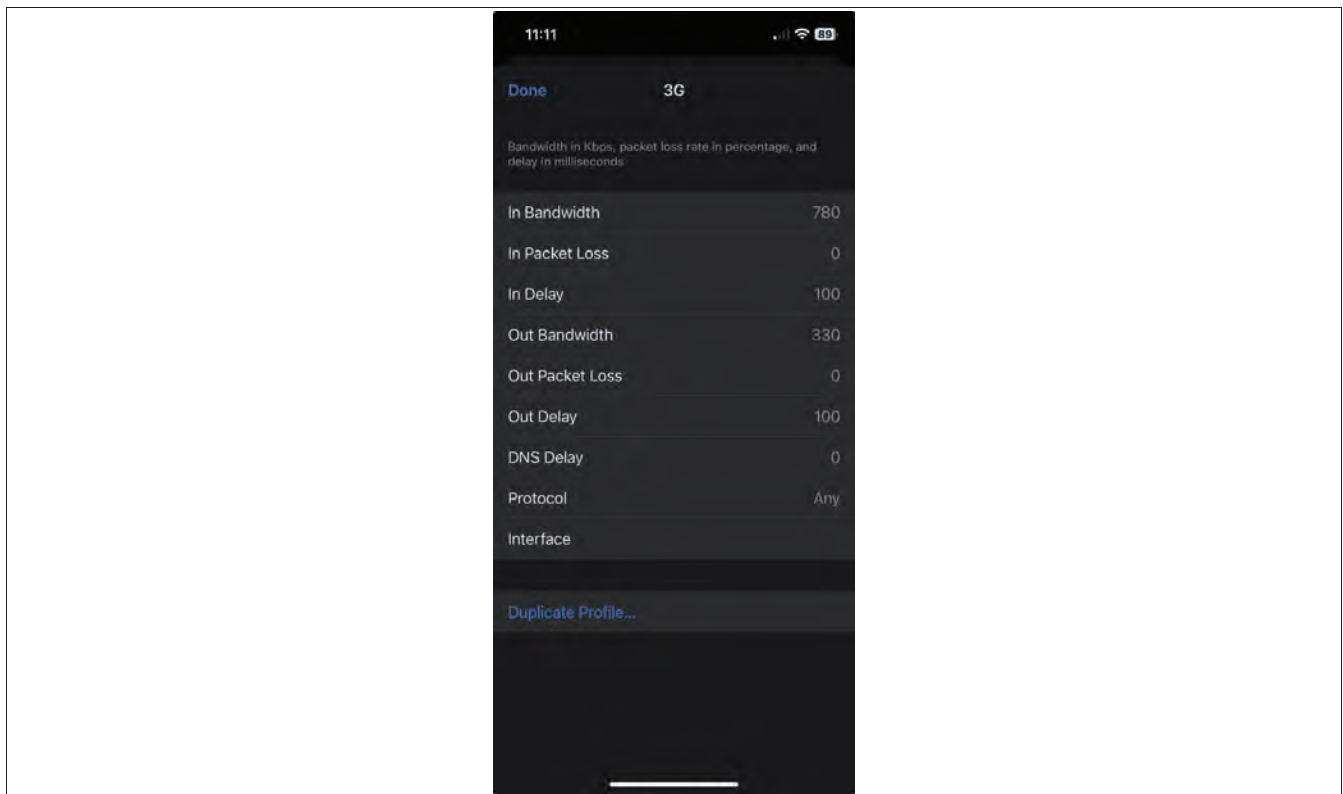
**[3] The system of claim 2, wherein the one or more connection simulations are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation.**

The Accused Instrumentalities are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation. For example, when a user selects a wireless network such as 3G or LTE in Network Link Conditioner, then Network Link Conditioner will simulate wireless transmission of content to the mobile device by simulating network characteristics that would occur when transmitting wireless content to the mobile device.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[3] The system of claim 2, wherein the one or more connection simulations are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation.

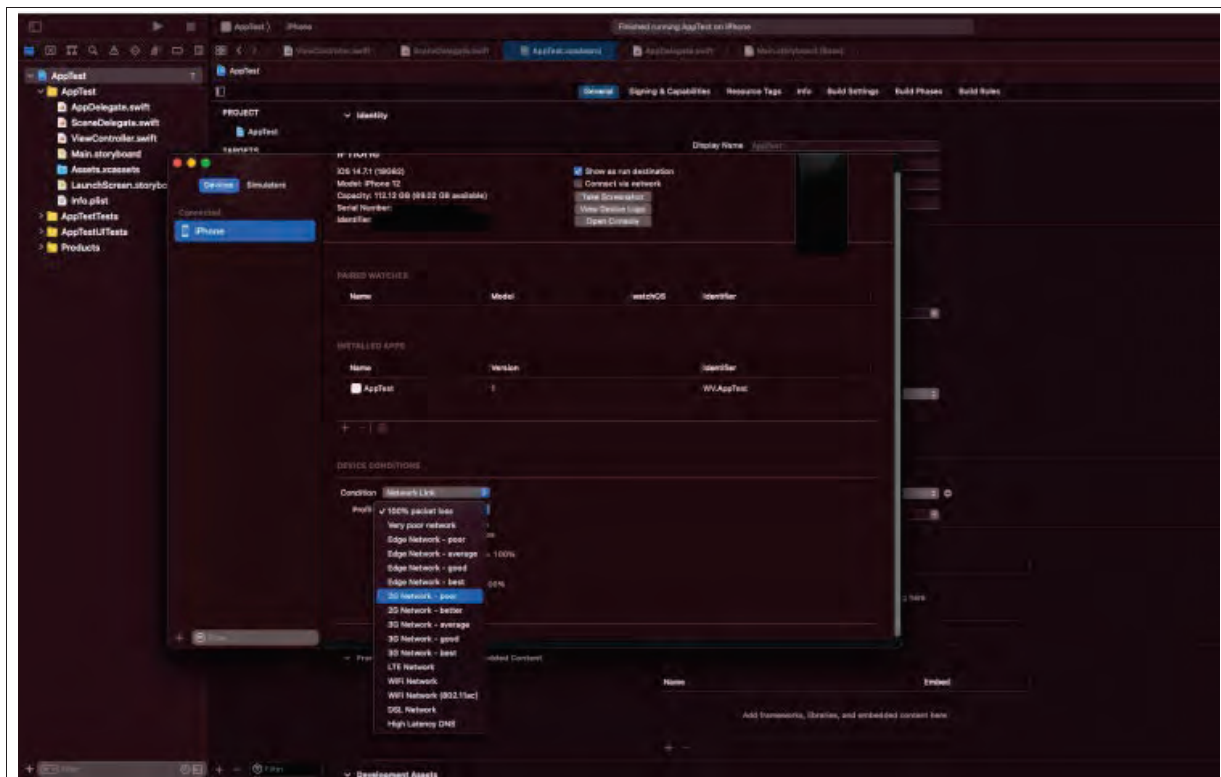
3G Setting for Network Link Conditioner on iOS:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[3] The system of claim 2, wherein the one or more connection simulations are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation.

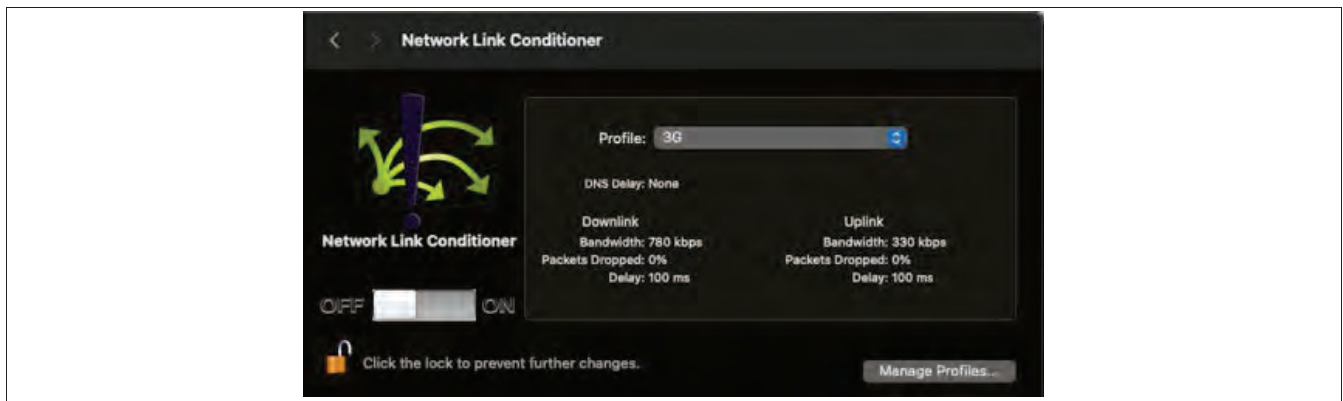
2G Setting for Network Link Conditioner on iOS configured through Xcode IDE:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[3] The system of claim 2, wherein the one or more connection simulations are configured to simulate wireless transmission of content to the mobile device based on the selected connection simulation.

3G Setting for Network Link Conditioner on MacOS:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[4] The system of claim 2, wherein the connection simulation includes one or more profiles.

**Claim 4**

**[4] The system of claim 2, wherein the connection simulation includes one or more profiles.**

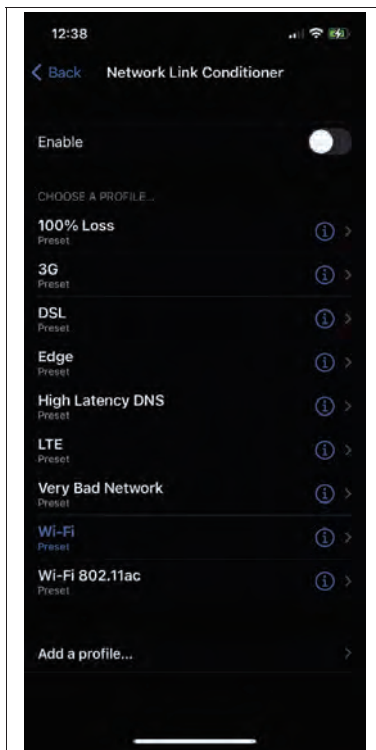
In the Accused Instrumentalities, the connection simulation includes one or more network profiles:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[4] The system of claim 2, wherein the connection simulation includes one or more profiles.

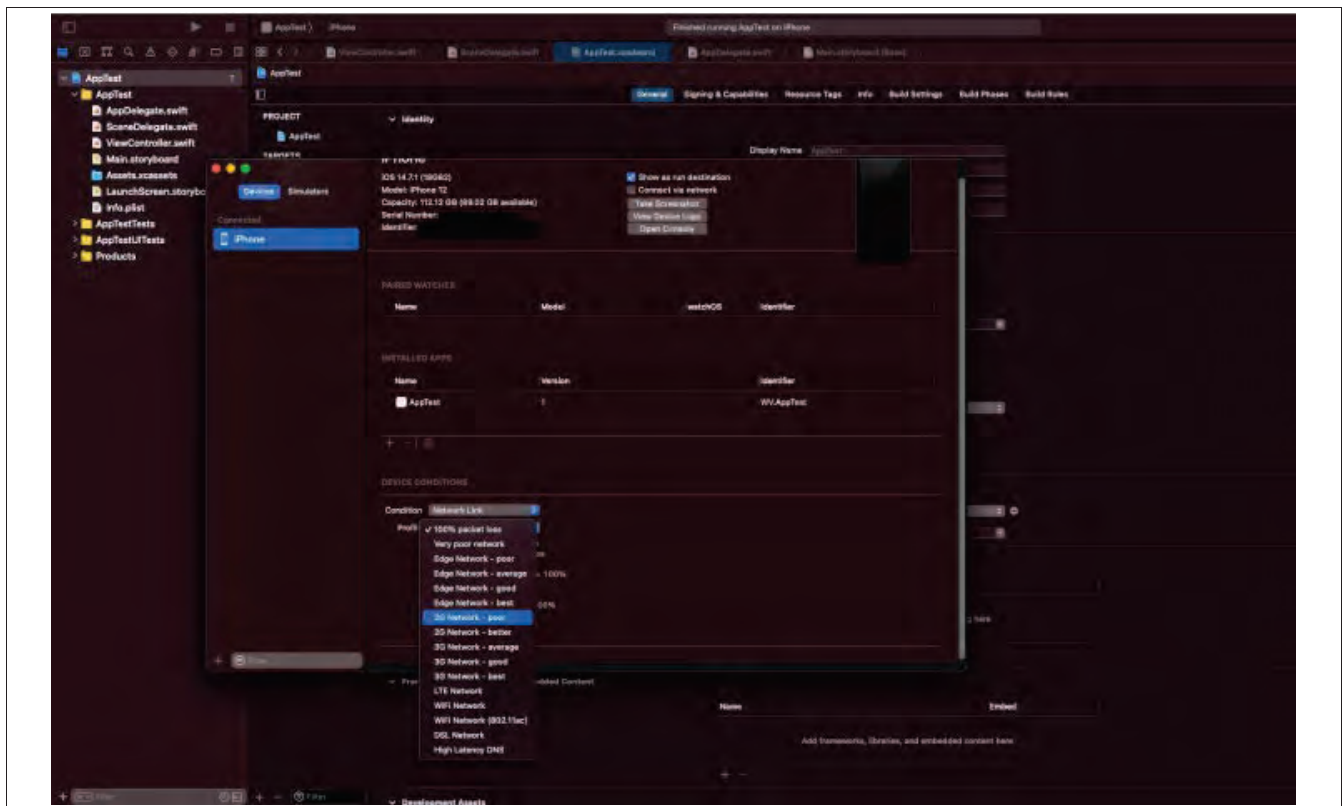
**1. Network Link Conditioner on iOS**



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[4] The system of claim 2, wherein the connection simulation includes one or more profiles.

Network Link Conditioner on iOS configured through Xcode IDE:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[4] The system of claim 2, wherein the connection simulation includes one or more profiles.

**2. Network Link Conditioner on MacOS**



The connection simulations include the profiles listed in Network Link Conditioner, such as: 100% Loss, 3G, DSL, Edge, High Latency DNS, LTE, Very Bad Network, Wi-Fi, Wi-Fi 802.11ac, Very poor network, Edge Network – poor, Edge Network – average, Edge Network – good, Edge Network – best, 2G Network – poor, 2G Network – better, 3G Network – average, 3G Network – good, 3G Network – best.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[5] The system of claim 4, wherein the profiles include preset profiles.

**Claim 5**

**[5] The system of claim 4, wherein the profiles include preset profiles.**

As shown above for Claim 4 (and incorporated here by reference), Network Link Conditioner profiles include multiple preset profiles that are included with Network Link Conditioner by default, such as: 100% Loss, 3G, DSL, Edge, High Latency DNS, LTE, Very Bad Network, Wi-Fi, Wi-Fi 802.11ac, Very poor network, Edge Network – poor, Edge Network – average, Edge Network – good, Edge Network – best, 2G Network – poor, 2G Network – better, 3G Network – average, 3G Network – good, 3G Network – best.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

**Claim 6**

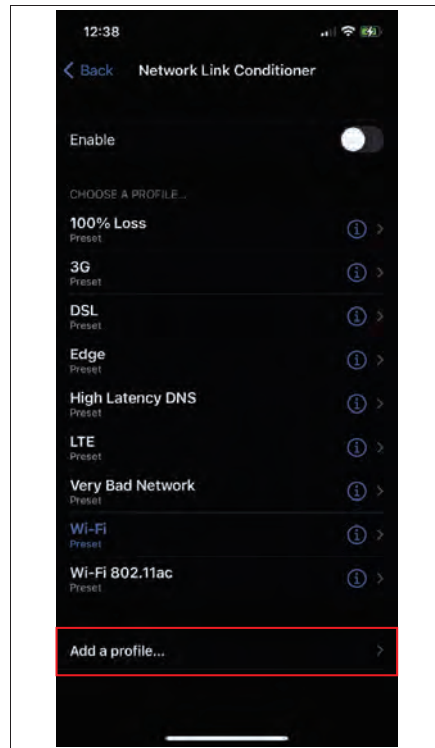
**[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.**

The profiles in the Accused Instrumentalities are configured to enable a user to manage the profiles, such as by creating new profiles, modifying previously created profiles, and deleting previously created profiles.

For example, on Network Link Conditioner on iOS users can create new profiles:

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

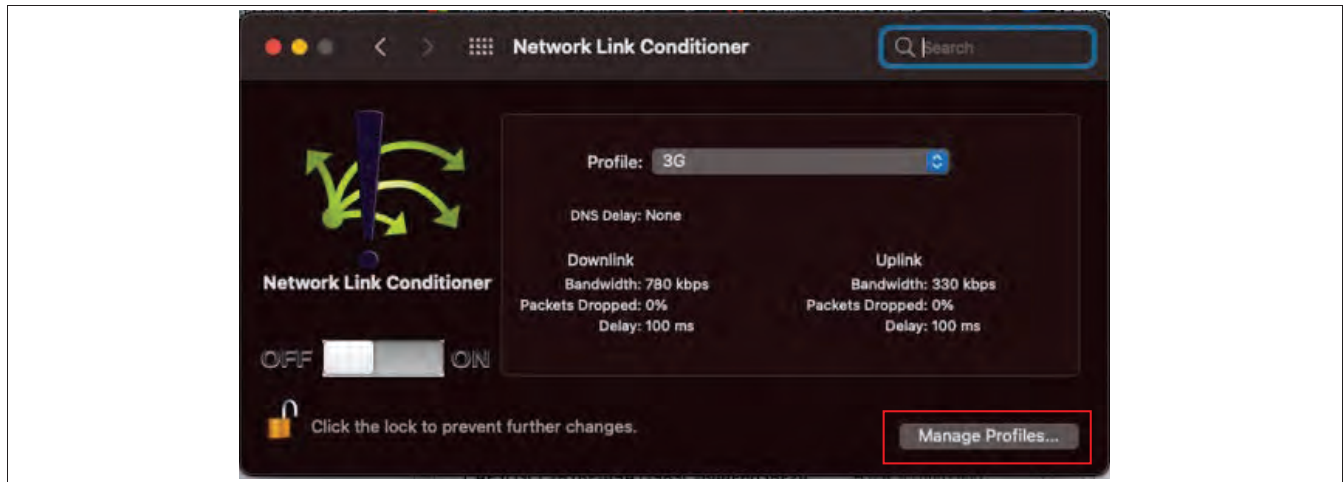
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

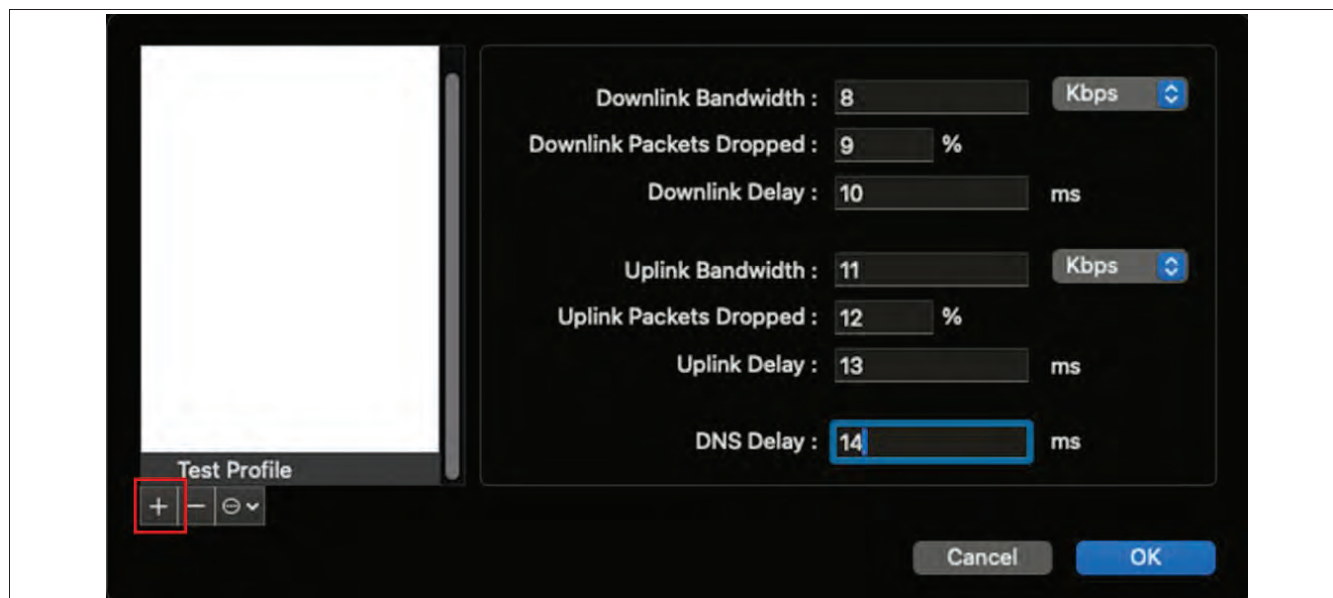
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

Similarly, users can create profiles on Network Link Conditioner on MacOS:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

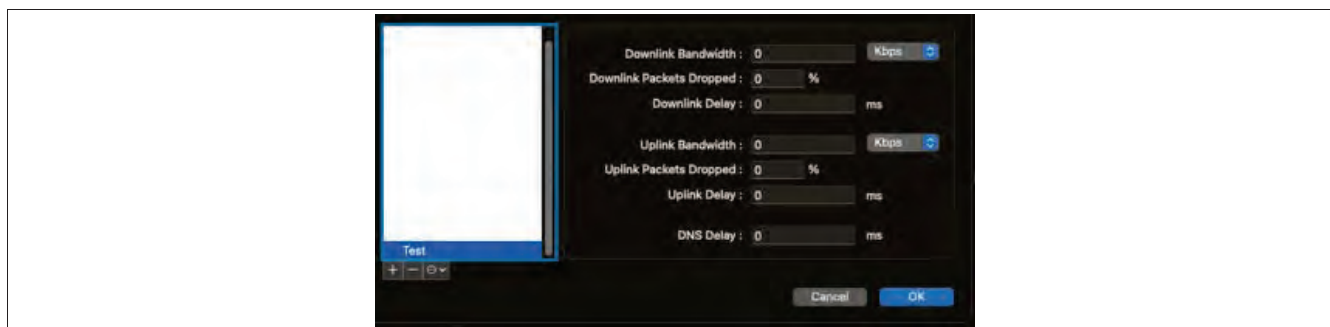
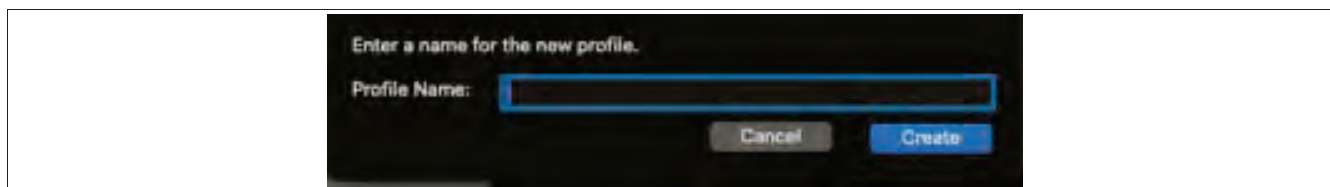
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.





**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

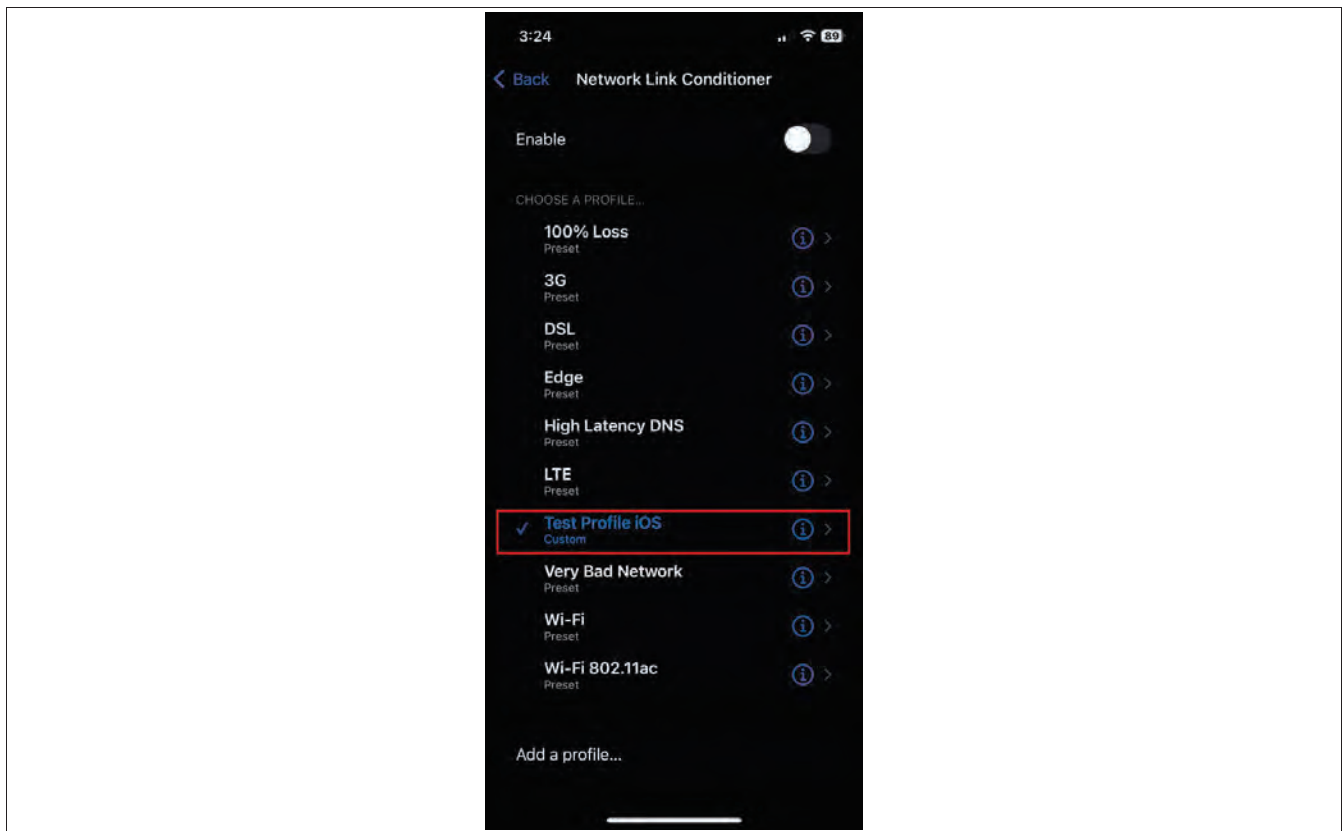
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

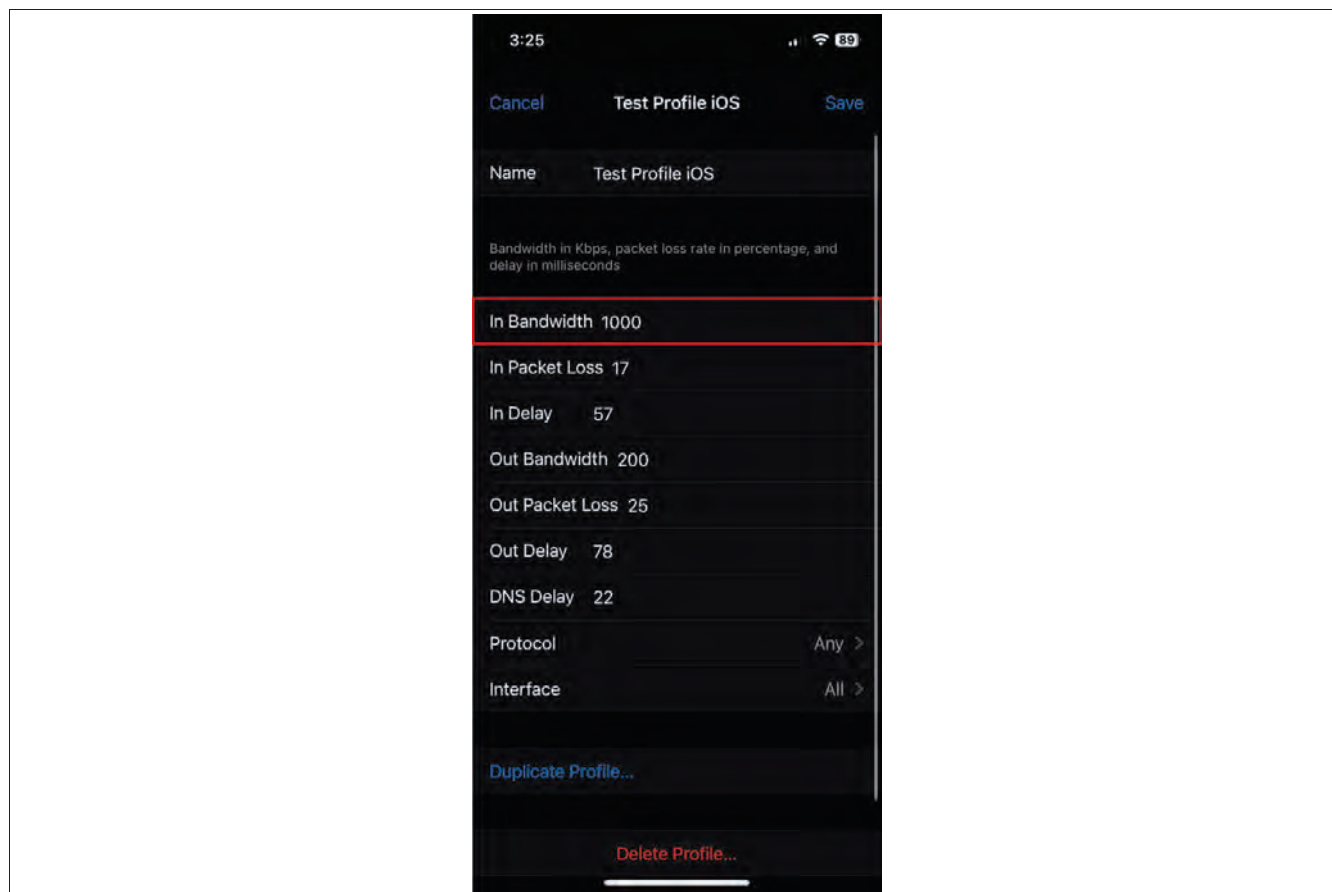
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

Users can also modify previously created profiles. Network Link Conditioner on iOS:



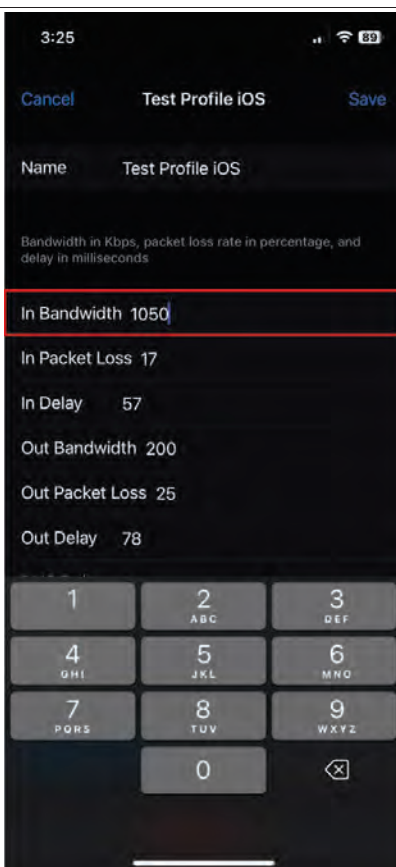
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

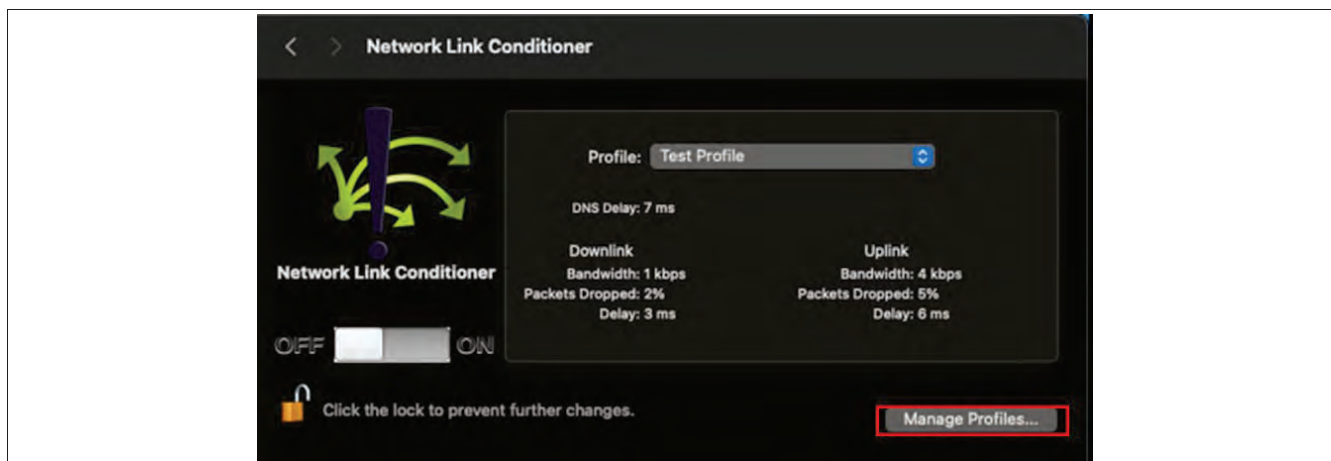
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

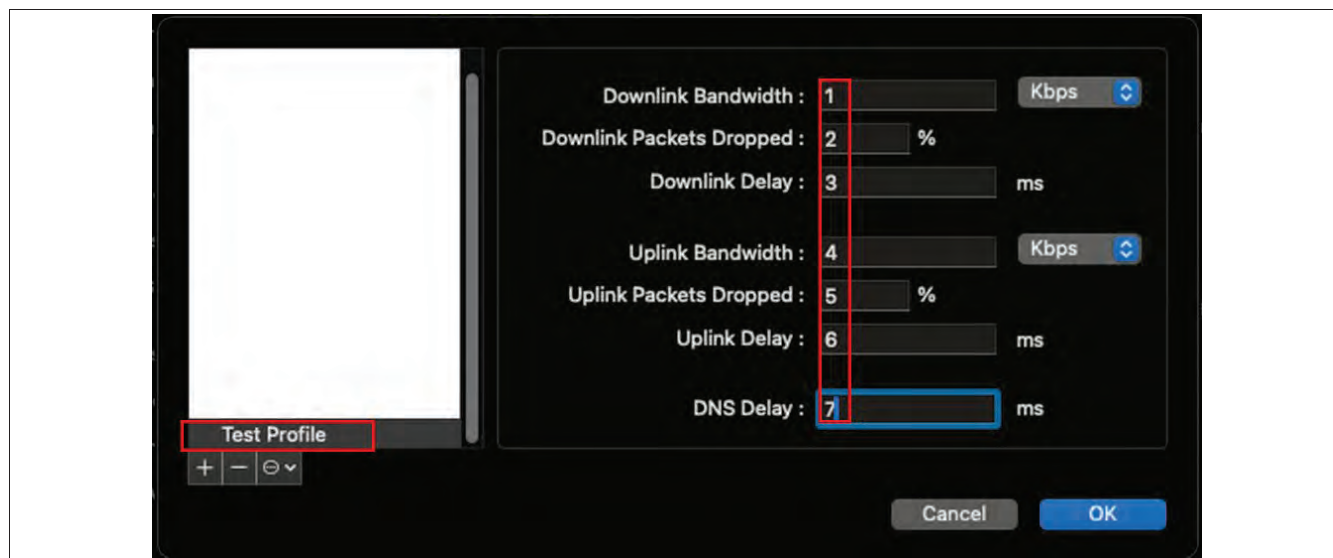
[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

Network Link Conditioner on MacOS:



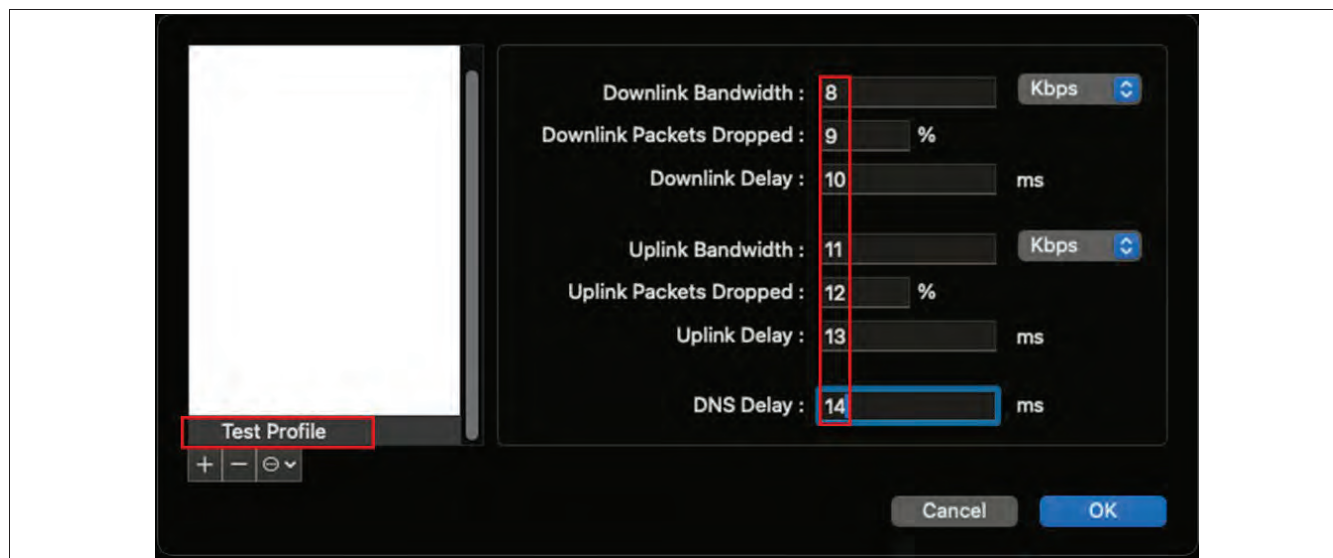
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.



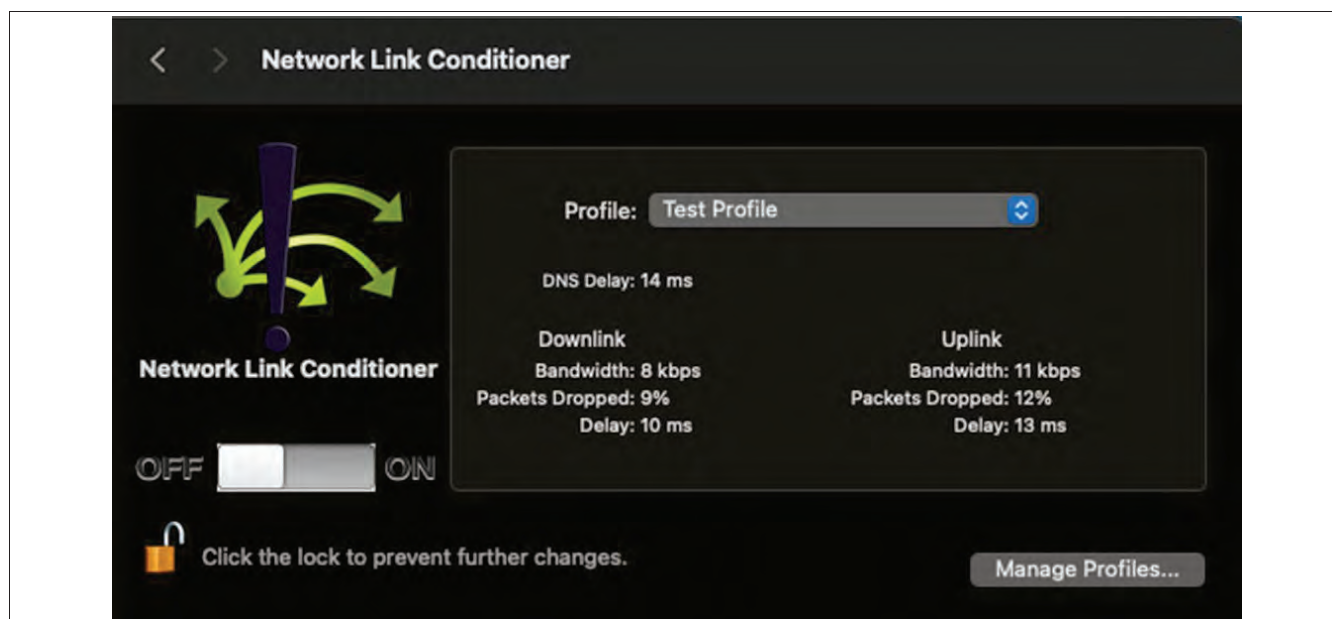
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

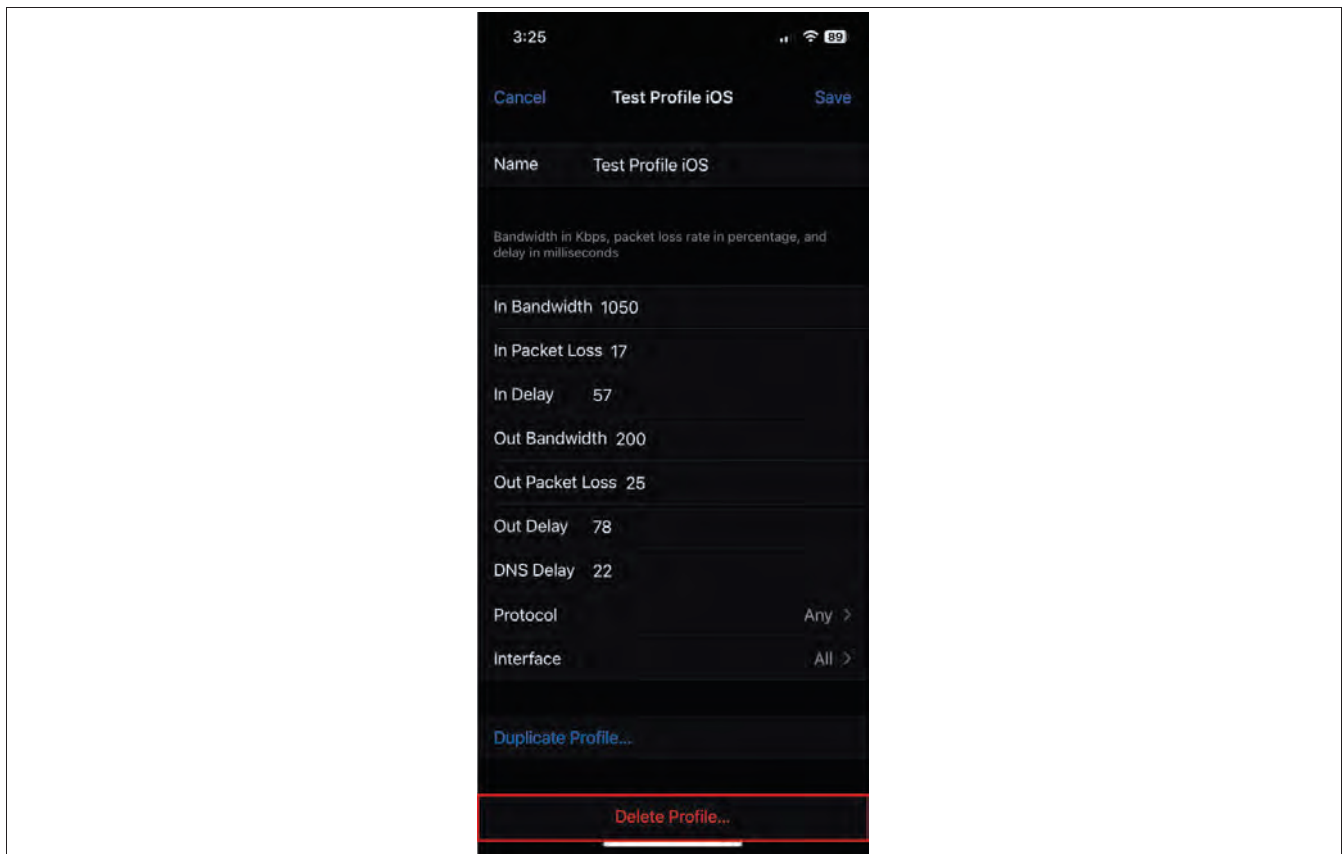




**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

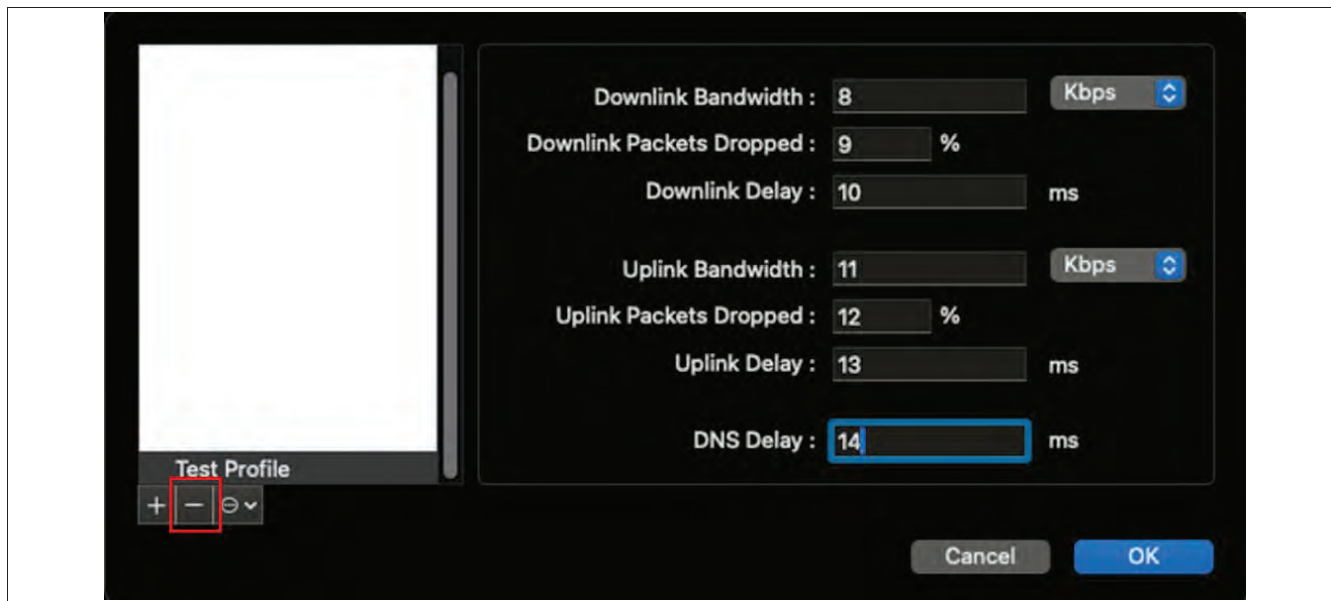
Users can also delete previously created profiles. Network Link Conditioner on iOS:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[6] The system of claim 4, wherein the profiles are configured to enable a user to manage the profiles.

Network Link Conditioner on MacOS:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[7] The system of claim 4, wherein the profiles are configured to enable a user to create custom profiles.

**Claim 7**

**[7] The system of claim 4, wherein the profiles are configured to enable a user to create custom profiles.**

As discussed above for Claim 6 (incorporated here by reference), the profiles are configured to enable a user to create custom profiles.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.

**Claim 9**

**[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.**

The connection simulations simulated by Xcode Network Link Conditioner are based on data of interaction with networks in non-simulated environments.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

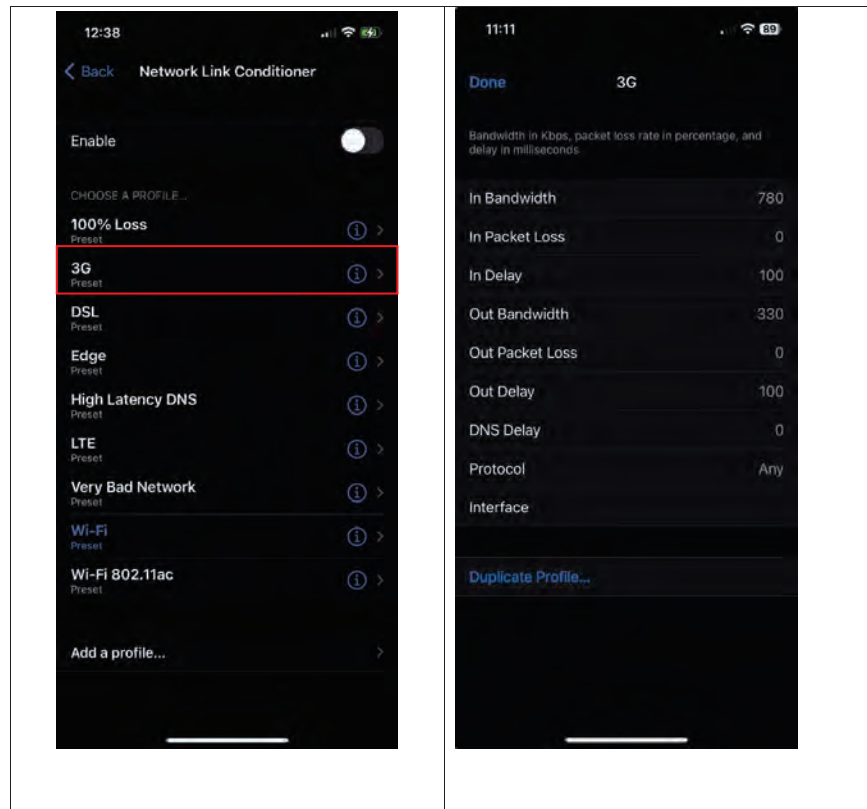
[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.



Network Link Conditioner on MacOS.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

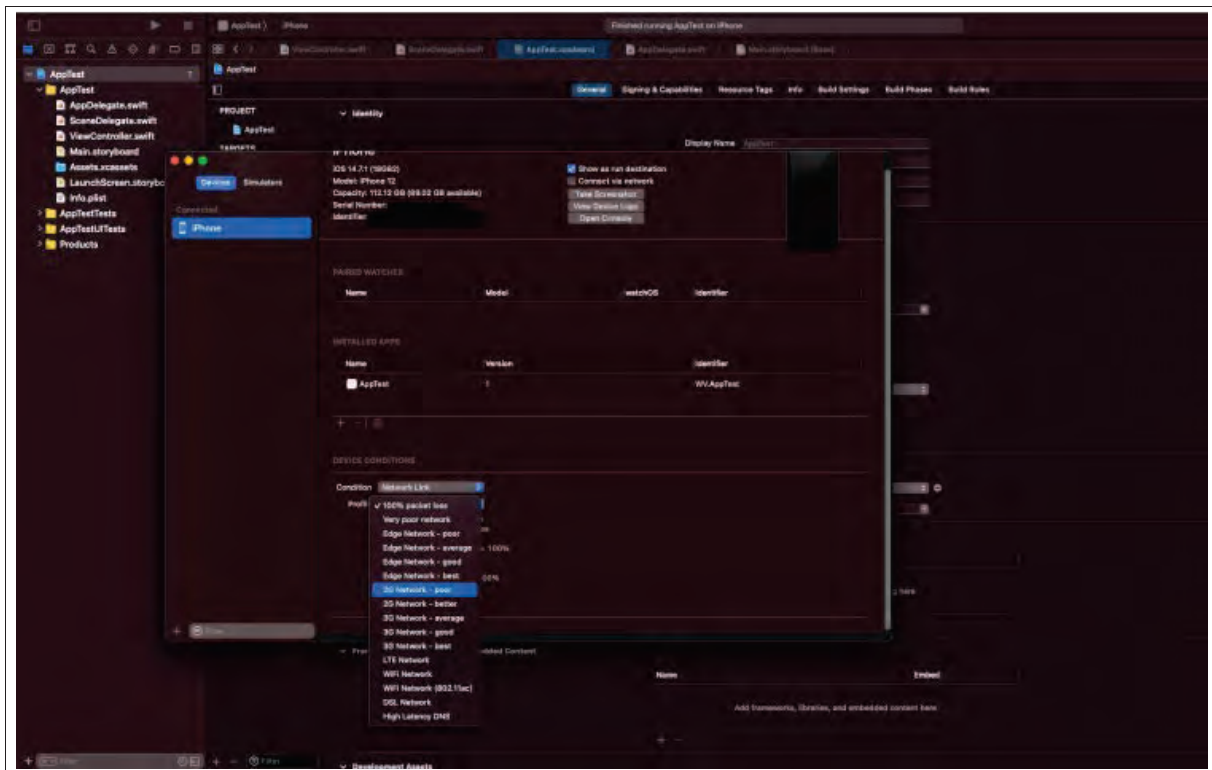
[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.



Network Link Conditioner on iOS.

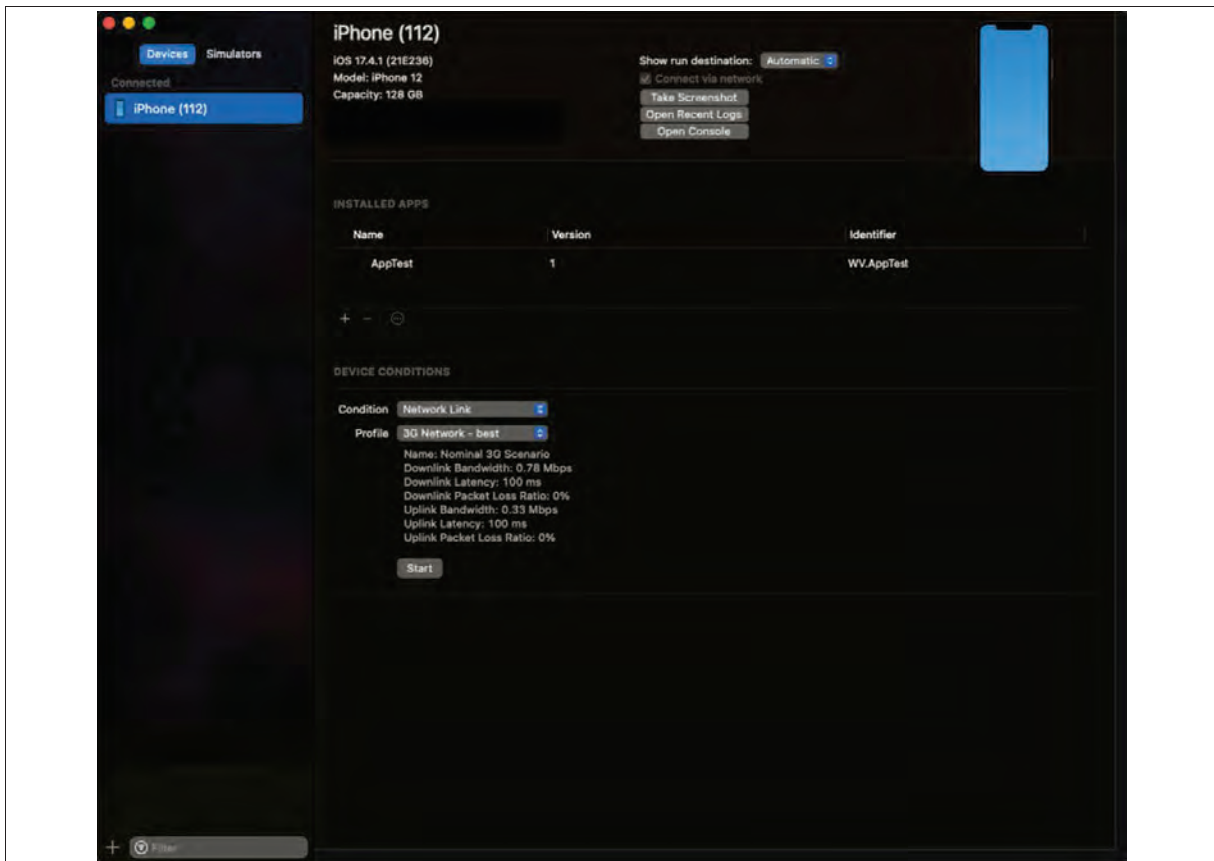
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.



Network Link Conditioner on iOS configured through Xcode IDE.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.

The network characteristics such as bandwidth, latency/delay, and packet loss are based on data of interactions with networks in non-simulated environments at least because of their reliance on network standards developed to dictate the operation of real-world networks, such as LTE, 2G, 3G, and EDGE.

For instance, GSM corresponds to what is often referred to as the “2G” network standard established around 1991. Similarly, EDGE corresponds to Enhanced Data rates for GSM Evolution, an enhancement to GSM. And HSCSD corresponds to High Speed Circuit Switched Data, which is an enhancement to the data rate of circuit switched data in a GSM network. GPRS corresponds to a packet-oriented enhancement to 2G networks—General Packet Radio Service. UMTS is the Universal Mobile Telecommunications System, a new architecture that provided the basis for what is often referred to as the “3G” network standards. HSDPA, or High-Speed Downlink Packet Access, is an enhancement to the 3G network architecture to boost data capacity and improve download rates. Finally, LTE, or Long Term Evolution, represents the transition from 3G to what is typically referred to as “4G” network technology.

Each of these network standards defines the general operation of the network, and these definitions provide theoretical constraints on the networks’ capacity for communication, including bandwidth, latency, packet loss, and speed constraints. These constraints can be further impaired based on network conditions, including the presence of physical obstacles, electro-magnetic interference, and/or distance between the base station and a mobile station with which it is communicating.

The development and evolution of these standards relied on data of interactions with real-world implementations of such networks at least for testing and proof-of-concept. Thus, Network Link Conditioner’s bandwidth, packet loss, and latency/delay constraints correspond to each identified standard, which are based on data of interaction of networks in non-simulated environments.

Many network operators have participated in (and continue to participate in) the cellular standards detailed above. This participation includes both influencing the direction of the standards as well as testing the viability or early implementation of proposed standards.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.

**a. The 3GPP Standards Development Process**

Verizon is participating in and influencing the 5G standards setting process through the 3rd Generation Partner Project (3GPP), which has previously provided LTE, LTE-Advanced and LTE Advanced Pro for commercial cellular/mobile systems. There are seven organizational partners in 3GPP which work on the standards and also several peripheral organizations that reference or provide input to 3GPP standards (Figure 1).

<https://www.verizon.com/business/resources/whitepapers/first-principles-for-securing-5g/> (last visited 5/17/2024); *see also id.* (“The standards development process, including work on security features, benefits from input from companies with real-world experience deploying new technology. It is common for companies like Verizon who are “first movers” to deploy service using new technology while the standards are still in development.”).

**AT&T Teams Up with Global Technology Leaders for Faster 5G Deployment**

AT&T\* is working with several global technology leaders and operators to align on 5G. The efforts are in preparation for the release of the official 3GPP specifications which will form the basis of the global standards.

....

In 5G trials, AT&T is accelerating over-the-air interoperability testing based on standards developed under the 3GPP New Radio (NR) specifications. The trials are designed to easily evolve with future versions of the official 5G standards, a milestone 3GPP targets for 2018. Focusing on the NR standards helps ensure the technology will work correctly with any future 3GPP 5G NR updates.

[https://about.att.com/story/faster\\_5g\\_deployment.html](https://about.att.com/story/faster_5g_deployment.html) (last visited 5/17/2024).

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.

Verizon has said that its goal is to collaborate with vendors on some early specifications and then contribute those to the 3GPP, the mobile industry standards body responsible for creating the 5G standard. Verizon insists that its early release of these 5G specifications won't create fragmentation in the industry.

....

When Verizon released its 5G spec last July, the company said the guidelines were primarily for testing and validating 5G components that will help chipset vendors and others develop interoperable 5G gear and assist with pre-standard testing and fabrication.

<https://www.sdxcentral.com/articles/news/verizon-att-spar-5g-standards-process/2016/09/> (last visited 5/17/2024).

When Network Link Conditioner was released, wireless networks had been widely deployed by network operators. On information and belief, Network Link Conditioner's bandwidth, packet loss, and latency/delay constraints are based on data of interaction with such networks in non-simulated environments operated by network operators.

These protocols are built for use in non-simulated environments, notably, operator networks. Therefore, Network Link Conditioner's connection simulations are based on data of interaction with network operators in non-simulated environments.

This is further reinforced by Apple's documentation, which instructs developers to use Network Link Conditioner to simulate "real-world conditions":

**Test Under Various Conditions**

Xcode provides a tool called Network Link Conditioner that can simulate various network conditions, including reduced bandwidth, high latency, DNS delays, packet loss, and so on. Before you ship any software that uses networking, you should install this tool, enable it, then run your software to see how it performs under real-world conditions.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[9] The system of claim 2, wherein the one or more connection simulations are based on data of interaction with network operators in non-simulated environments.

<https://developer.apple.com/library/archive/documentation/NetworkingInternetWeb/Conceptual/NetworkingOverview/WhyNetworkingIsHard/WhyNetworkingIsHard.html>, accessed on March 13, 2024.

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[21] The system of claim 1, wherein the software is further configured to display data to identify application performance.

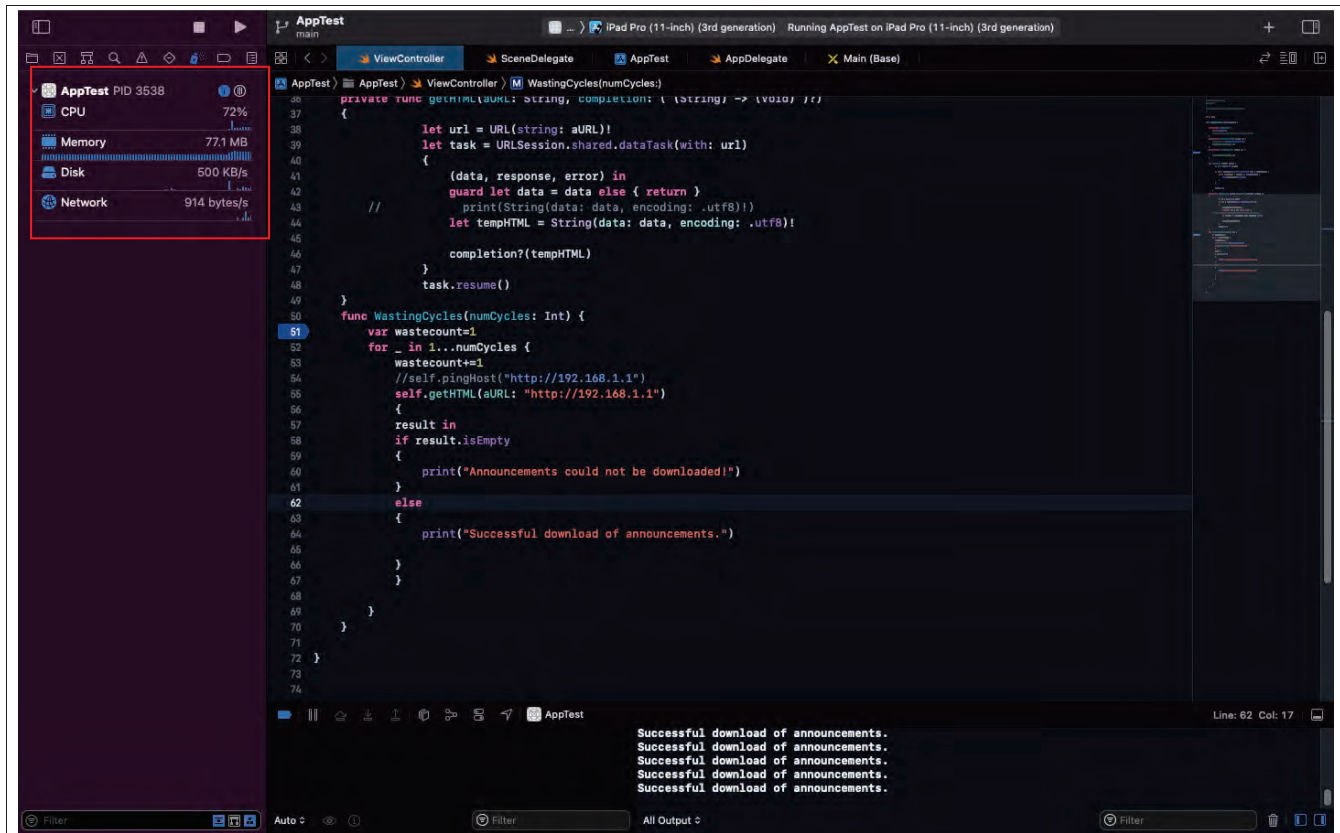
**Claim 21**

**[21] The system of claim 1, wherein the software is further configured to display data to identify application performance.**

As discussed above for limitation [1a].2 (and incorporated here by reference), the software is further configured to display data to identify application performance. It does this through the data displays found in Xcode IDE and Instruments, including the examples shown below (and discussed in additional detail for limitation [1a].2):

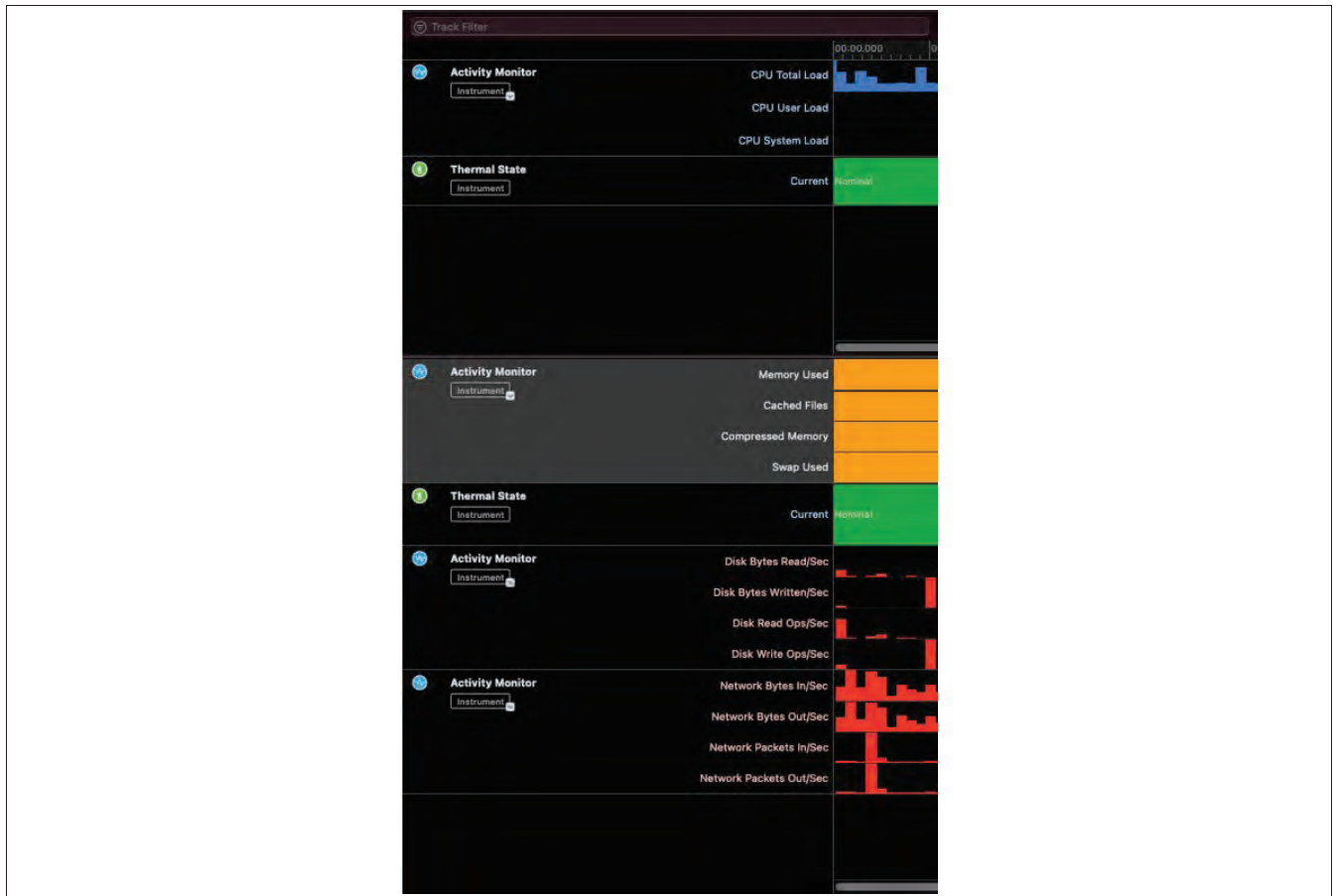
**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[21] The system of claim 1, wherein the software is further configured to display data to identify application performance.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[21] The system of claim 1, wherein the software is further configured to display data to identify application performance.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[21] The system of claim 1, wherein the software is further configured to display data to identify application performance.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[22] The system of claim 21, wherein the user can manage the data.

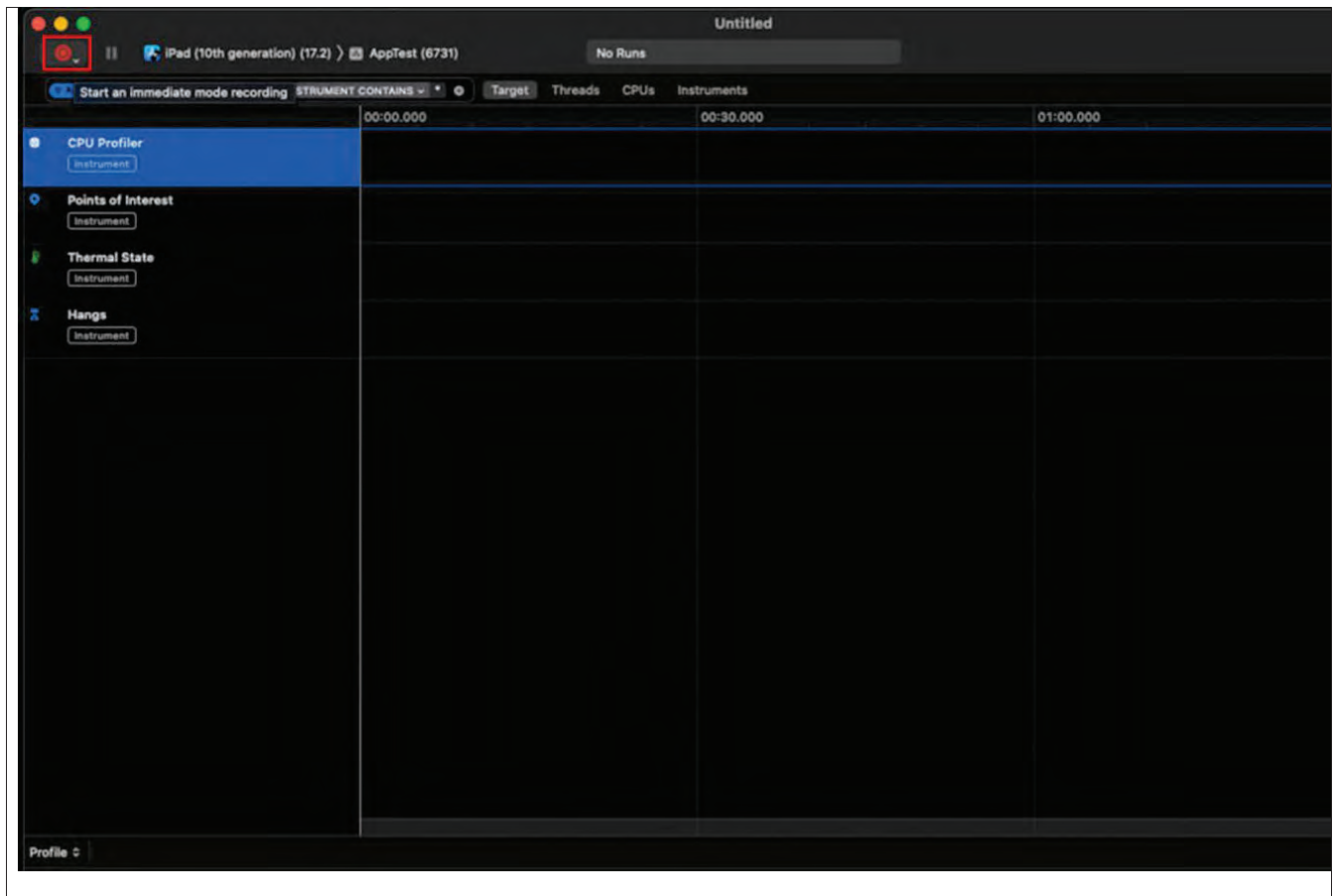
**Claim 22**

**[22] The system of claim 21, wherein the user can manage the data.**

Users of the Accused Instrumentalities can manage the data displayed to identify application performance, for example by saving and loading the data. As discussed previously, this data can be captured and displayed by, for example, clicking the “record” button within Xcode instruments (annotated in red below):

**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

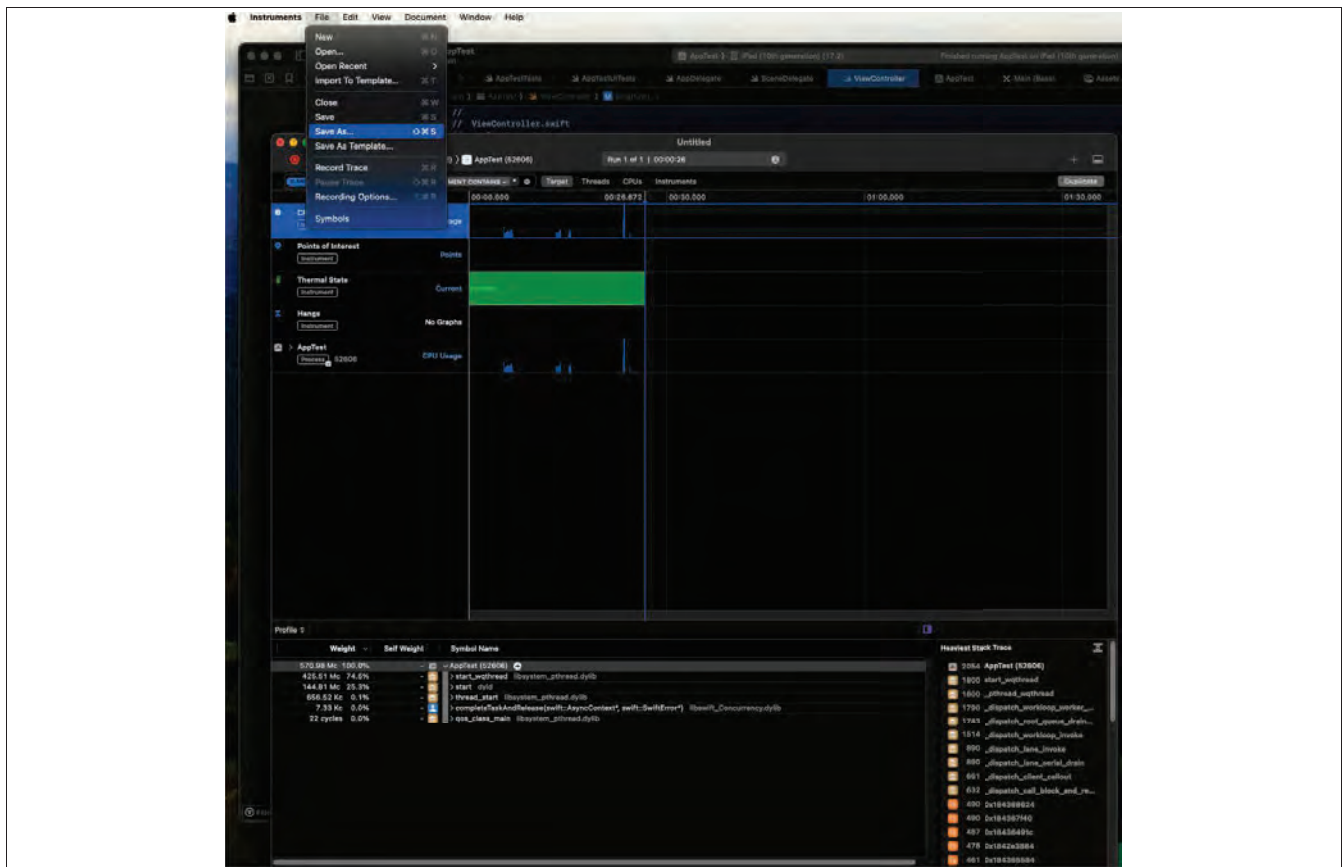
[22] The system of claim 21, wherein the user can manage the data.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

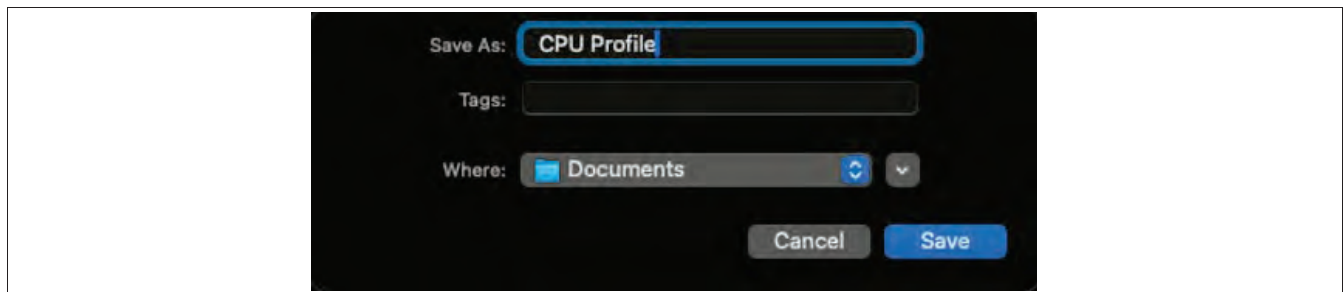
[22] The system of claim 21, wherein the user can manage the data.

After being collected, this data can be saved for later analysis:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

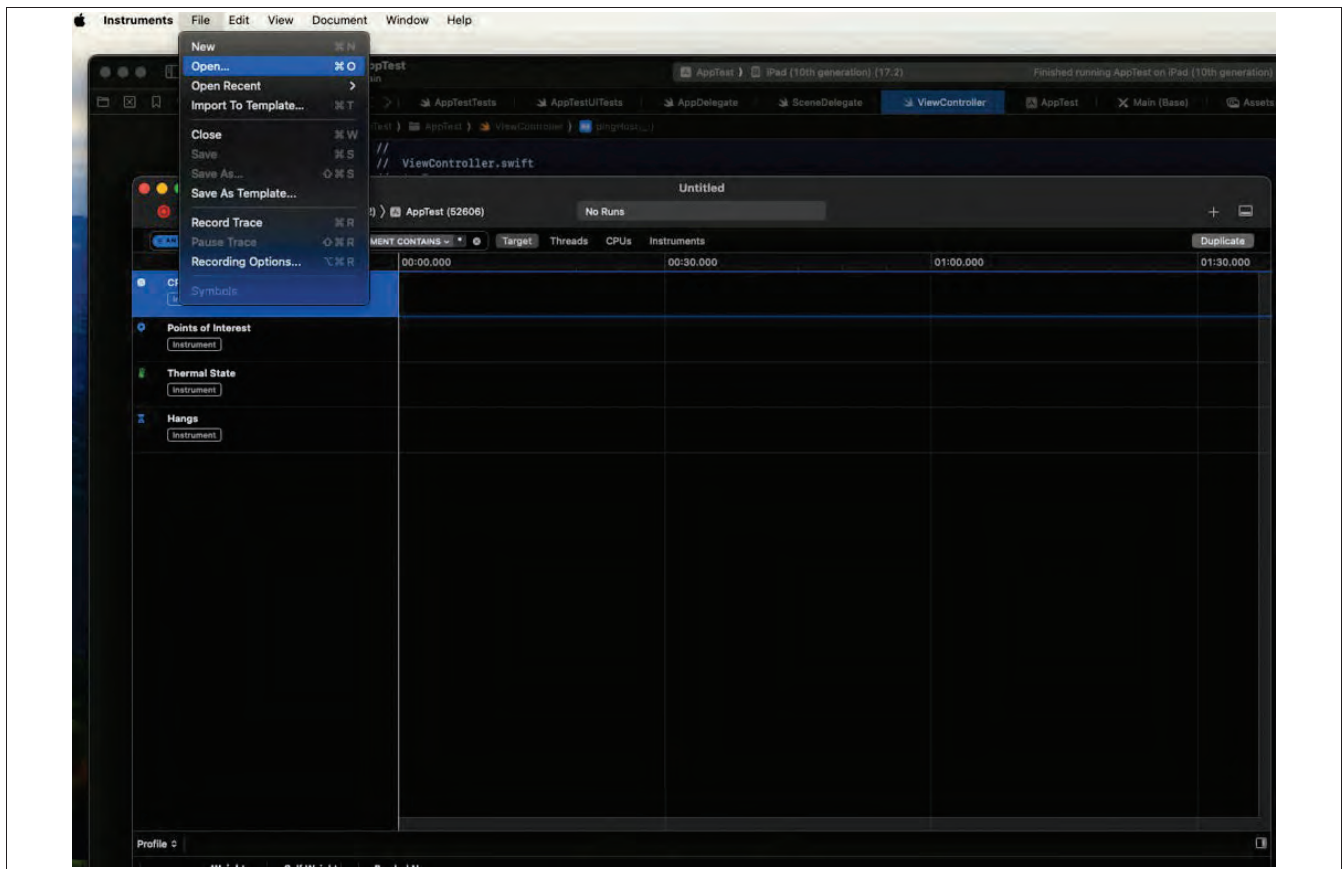
[22] The system of claim 21, wherein the user can manage the data.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

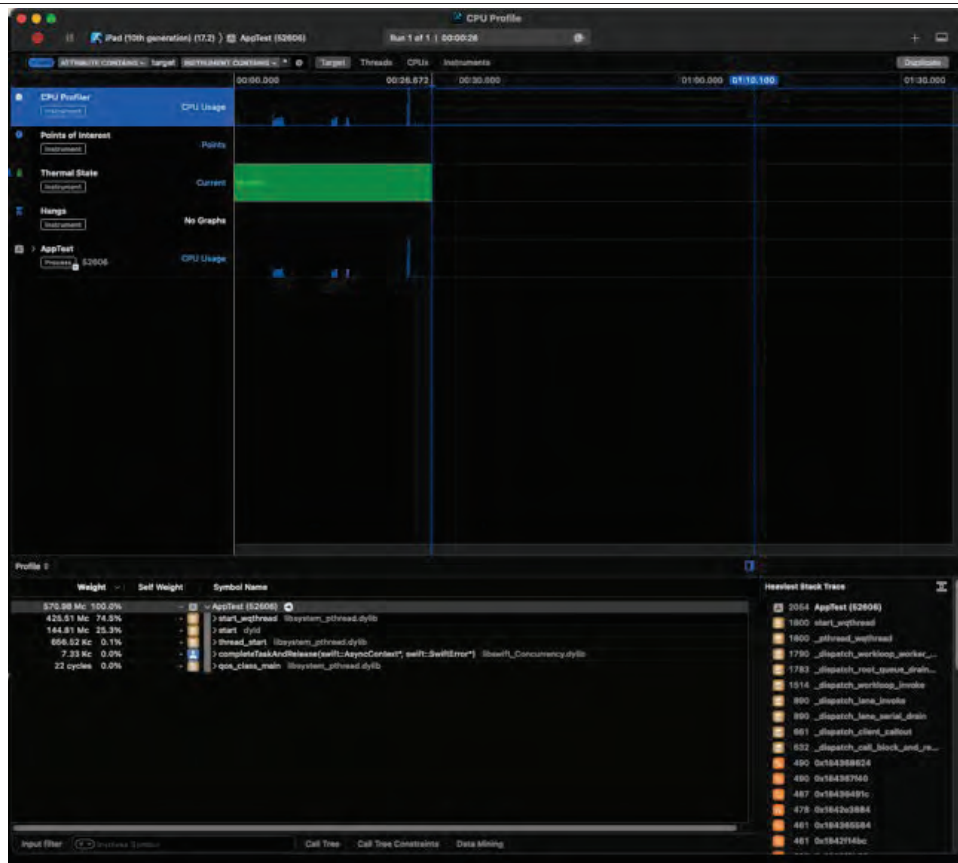
[22] The system of claim 21, wherein the user can manage the data.

Previously collected data can similarly be loaded for analysis:



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[22] The system of claim 21, wherein the user can manage the data.



**Infringement Chart for U.S. Patent No. 9,971,678 (Apple's Xcode Developer Tools)**

[22] The system of claim 21, wherein the user can manage the data.